AR-009-481

DSTO-GD-0077

Proceedings of the Artificial Intelligence
in Defence Workshop AI'95

Edited by Simon Goss

APPROVED FOR PUBLIC RELEASE

© Commonwealth of Australia

DEPARTMENT ◆ OF DEFENCE

DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION

Proceedings of the Artificial Intelligence in Defence Workshop

AI'95

Edited by Simon Goss

Eighth Australian Joint Conference on Artifical Intelligence

Australian Defence Force Academy
Canberra
14 November 1995

DSTO-GD-0077

DTIC QUALITY INSPECTED 3

ABSTRACT

A workshop with the theme "Artificial Intelligence in Defence" was held at the Australian Joint Conference on Artificial Intelligence at the Australian Defence Force Academy in November 1995. There were 52 attendees from defence, defence science, industry and academia. Twelve papers were presented in four thematic areas: Decision Support, Surveillance and Information Fusion, Modelling and Operations Research, and Simulation and Training. This proceedings documents the final versions of those papers.

RELEASE LIMITATION

Approved for public release

DEPARTMENT OF DEFENCE

◆

DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION

INTRODUCTION

This proceedings collates the papers presented at the AI in Defence Workshop held as part of the workshop series associated with AI-95, the joint national conference on artificial intelligence, held at the Australian Defence Force Academy in November 1995. The workshop was also one of the biannual meetings of the DSTO Special Interest Group in Artificial Intelligence. There were 52 attendees from defence, defence science, government, industry and academe.

The workshop aim was to focus on the deployment of AI in defence applications, and to promote interaction between the Defence, Defence Science and AI communities. Defence is a multi-faceted domain. Artificial Intelligence (AI) can effect significant improvement in performance from the "teeth" end of battlefield assistance through to the "tail" of routine logistical support. A successful fielded AI application providing defence benefit in the supporting the work of OR analysts, engineers and Materiel Command is defence applications is DRAIR ADVISER [1]. Defence in depth relies on information; information technologies have a substantial part to play. Data fusion and surveillance are areas in which AI techniques married to other technologies can have considerable impact on Defence effectiveness and efficiency.

The program of papers assembled here meets these aims and documents a day of engaging, stimulating and rewarding interchange. The eleven papers selected by the referees from those offered fall into the broad categories of:

Decision Support
> covering Human-Computer Interaction, C3I, and
> tools for threat assessment and hypothesis management,

Modelling, and Operations Research
> verification and validation of complex model, and
> the architecture of a complex model,

Surveillance and Information Fusion
> several perspectives in image interpretation and analysis,
> and in the management and presentation of multimodal
> information, and

Simulation and Training
> Computer Generated Forces.

I would like to thank Tracy Truong of DSTO Air Operations Division for assistance in assembling the final versions of the papers, the program committee, the attendees and, not least, the authors.

Simon Goss
Workshop Chair

(1) Brian L. Robey, Pamela K. Fink, Sanjeev Venkatesan, Carol L. Redfield, and Jerry Ferguson. (1994) DRAIR ADVISER: A Knowledge-Based System for Material Deficiency Analysis, AI Magazine 15 (2) pp 67-82.

Committee

Chair   Dr Simon Goss (DSTO-Air Operations Division)
        Dr Greg Gibbon (DSTO-Information Technology Division)
        Mr Graeme Murray (DSTO-Air Operations Division)
        Dr Sabrina Sestito (DSTO-Air Operations Division)
        Dr Grahame Smith (CRC in Intelligent Decision Systems)

Program Committee

Prof Terry Caelli (Curtin University)
Dr Vic Ciesielski (Royal Melbourne Institute of Technology)
Dr Jennie Clothier (DSTO Information Technology Division)
A Prof Paul Compton (UNSW)
Mr Graham Forsyth (DSTO Engines and Airframes Division)
Dr Greg Gibbon (DSTO-Information Technology Division)
Dr Simon Goss (DSTO Air Operations Division)
Dr Todd Mansell (DSTO Maritime Operations Division)
Mr Graeme Murray (DSTO Air Operations Division)
Prof Charles Newton (ADFA)
Dr Sabrina Sestito (DSTO-Air Operations Division)
Dr Grahame Smith (CRC in Intelligent Decision Systems)
Mr Gil Tidhar (Australian Artificial Intelligence Institute)

TABLE OF CONTENTS

Introduction

Interactive Planners and Human-Computer Interaction

Conn V Copas
HCI Lab
Information Technology Division
DSTO
e-mail: cvc@itd.dsto.gov.au

Abstract

Recent progress in planning has enabled this technique to be scaled-up to some significant real-world problems, including the construction of software agents. This paper examines this development from a perspective of human-computer interaction, with the reference domain being military spatial information processing, supported by a geographic information system. Work in interactive planners has emphasised their dynamism and maintenance advantages. This paper explores the theme that a paradigm shift in human-computer interaction is now a prospect: away from the requirement to instruct machines towards a more declarative, goal-based form of interaction. This initiative necessarily involves consideration of the design of goal description languages, and some alternatives are analysed. Some additional demands posed by the requirement to embed planners within user interface management systems are also examined.

1. Introduction

Recent progress in domain-independent planning research has allowed this technique to begin to realise some of its long-standing potential. For example, conditional action effects within partial-order planners were generally considered to be problematic until (Pednault, 1988) and still dubious in a formal sense until (Penberthy & Weld, 1992). It is only more recently still that robust techniques have been reported for incorporating desirable features such as disjunctive preconditions and quantification over dynamic object universes (Weld, 1994).

Quasi real-time planners are now being reported which make increasingly less restrictive assumptions; namely, that the planner has access to all necessary information about the state of the world, that exogenous events do not cause that state to change, and that action effects are both instant and deterministic. Whilst these restrictions may be regarded as unreasonable within certain real-
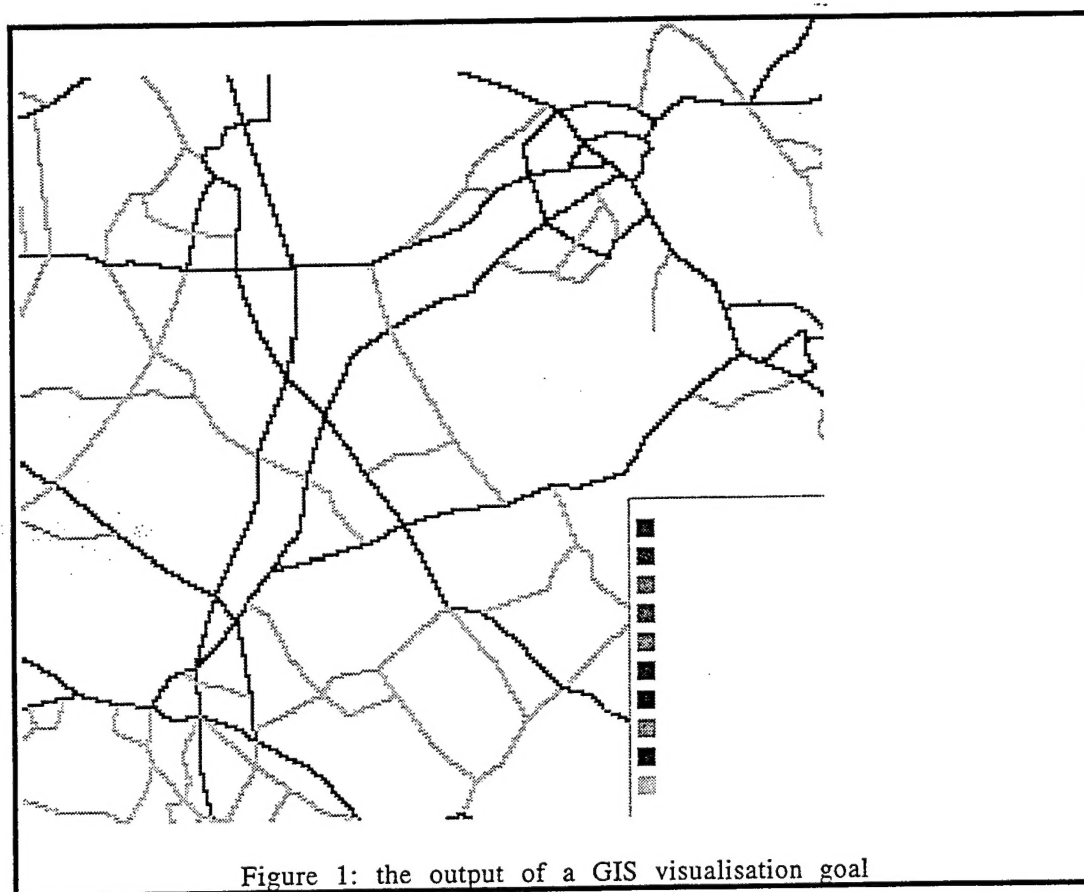
world domains, these restrictions may be quite reasonable in the case of the artificial world of software agents. The general principle is that these agents are the recipients of goals which describe some desired state(s) of a computer-based system. These agents have available to them various actions, which typically correspond to user-level commands, and possess knowledge about both the preconditions and effects of these commands. The planning task is to search for appropriate combinations of commands which projection suggests will achieve the goal. From a programming perspective, the plan is built at run-time, as opposed to the procedural approach of enumerating all the contingent command combinations within the program. Domains in which such agents have been constructed include network searching within the Unix operating system (Etzioni & Weld, 1994) and image processing (Chien, 1994). As these agents may be said to possess both effector-like and sensor-like actions, they have also been described as 'softbots' (Etzioni, 1993).

These developments are sufficiently novel that it is considered to be a useful function for this paper to report on the feasibility of employing this approach within a domain of military relevance: user interaction with a geographic information system (GIS). These systems, along with many other so-called high-functionality systems (Fischer, 1991) have a poor reputation for useability. As discussed in section 2, conventional engineering solutions to this problem, such as the construction of graphical user interfaces, suffer from inherent limitations which a software agent may overcome. The work on softbots has so far emphasised the benefits of dynamism and maintenance which these provide in comparison to 'agents' which operate in a more procedural fashion, and has only addressed end-user concerns indirectly. Accordingly, a second aim of this paper is to articulate some human-computer interaction (HCI) issues, with particular reference to the design of goal description languages.

The final aim of this paper is to situate these developments in interactive planners within the pragmatic context of user interface management systems (UIMS). These systems provide a structured framework in which to build interactive software, and have implications for the roles and functional capabilities of any embedded planner.

2

2. GIS User Interfaces

Consider the following task facing some GIS users, which will be used for illustration throughout the remainder of this paper. The system includes a number of data files, representing roads, elevation, population, etc, with the display currently being blank. The task primarily involves visualisation, and the users' desire could be paraphrased as follows: "I would like to see the roads data in plan view, superimposed upon a white background, containing a legend in the bottom right corner and a scale-bar in the top centre". The expected output of the system is depicted in Figure 1.

Figure 1: the output of a GIS visualisation goal

It may be objected that this task is undemanding, as it does not involve any particular sophistication in spatial analysis on the part of the user. However, it is a good example for precisely that reason, because even users who have a clear idea of their goals must still translate those goals into a series of GIS instructions which are both syntactically correct and semantically coherent. Under the command-driven interface of the public-domain GIS Grass4.1, the necessary sequence of instructions involves seven steps, as depicted in Figure 2.

```
d.mon   start=x0
d.erase   color=white
d.rast  -o  map=roads
d.scale  at=0,0
d.frame  frame=frame0  at=0,40,75,100
d.erase   color=black
d.legend  map=roads
```

Figure 2: a typical GIS command sequence, or plan

As may be inferred from Figure 2, GIS tend to possess a large, relatively primitive command-set out of a concern for general-purpose capability and, in that respect, resemble the Unix operating system. One response to this situation is the construction of graphical interfaces which at the least reduces the potential for syntactic errors, and preferably involves the identification of some higher-level, albeit task-specific, macros. A state-of-the-art, commercial GIS interface is shown in Figure 3.
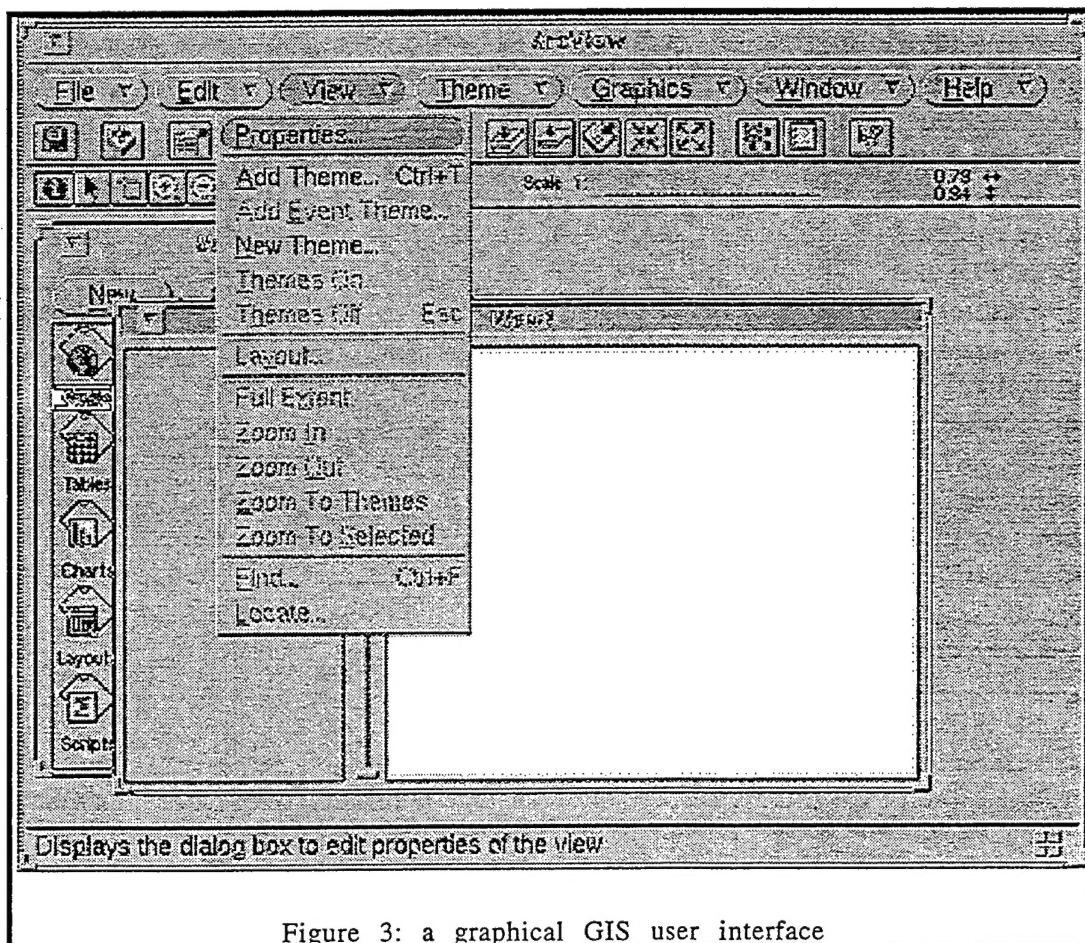

Figure 3: a graphical GIS user interface

The style of interface in Figure 3 has some obvious attractions in comparison to any command-line, but is itself not beyond modification. Typically, a further advance would be to supply some

iconic representation of the objects which comprise the system's universe of discourse, so that a direct manipulation style of interaction may occur. Within the GIS sphere, systems which incorporate such features are still at the experimental stage. One example is the employment of a cartographic overlay metaphor for combining maps visually (Egenhofer & Richards, 1993); another is the use of graphical pipelines to convey sequences of data transformations (Scopigno et al, 1990). However, if the experience within office systems is indicative, these developments may be expected to be limited ultimately by the problem of representing all available system actions (particularly abstract actions) in a gestural or pictorial fashion. Thus, for the medium term at least, user interfaces may be expected to place heavy reliance on the type of pop-up and pull-down menus of Figure 3, regardless of whether that interface is 2D, 3D or virtual.

One significant feature of these menus is that, linguistically, the items are almost invariably imperatives and, in the simplest case, correspond to application commands. Thus, the influence of the command-line lingers; in fact, it could be argued that the imperative language in which most systems are programmed has permeated through to the user interface, despite the best efforts of designers to construct various facades. It is at this point that developments in planning offer an alternative. Interactive planners are not simply 'intelligent' because, from an end-user perspective, this quality is largely a matter of functional capability. Interactive planners may be distinguished from other agents by their provision of a goal-centred or state-based form of interaction which is inherently more declarative than procedural. This is no accident, of course, as most planners derive from a logical foundation of predicate calculus and non-deterministic search. It is slightly ironic that, if planning technology becomes sufficiently well-understood to be appropriated by the mainstream (in the manner of the relational calculus, for example), then these agents may be regarded as routine constraint satisfiers!

Thus, a prospect which has been tantalising for some time is closer to realisation: users, instead of issuing numerous instructions in order to achieve their goals, may instead interact with machines by describing their goals in terms of attributes. One assumption underlying the efficacy of this approach is that the goal-set is smaller or at least more concise than any instruction-set which achieves those goals; otherwise, an imperative style of interaction becomes more attractive.

3. An Interactive Planner for GIS

The work reported here employs the public-domain (and domain-independent!) planner UCPOP2 (Weld, 1994), written in Common Lisp. This is a regressive, partial-order planner which has the features and limitations described in section 1. The public-domain release is non-hierarchical, although it plans abstractly in the sense of delaying commitment to variable bindings. Extensions have been reported for incorporating hierarchical reasoning (Barrett & Weld, 1994) and exogenous changes to world state have been addressed in a preliminary fashion (Etzioni et al, 1994).

The visualisation goal described in section1 is represented using existentially-quantified, first-order predicates and UCPOP2 syntax in Figure 4.

```
:goal    (exists (window ?x)
          (exists (frame ?y)
          (exists (scale-bar ?z)
          (and
(background-colour ?x white)
(displayed-in ?x map roads)
(contains ?x ?y)
(position frame ?y "0 40 75 100")
(displayed-in ?y legend roads)
(displayed-in ?x scale-bar ?z)
(position scale-bar ?z "0 0") ))))
```

"I would like to see the roads data in plan view, superimposed upon a white background, containing a legend in the bottom right corner and a scale-bar in the top centre."

Figure 4: A GIS goal, expressed in terms of both predicate logic and natural language

Confirmation that this goal is non-trivial (at least to any planner) comes from an examination of the expressiveness required to model adequately one of the GIS commands, as shown in Figure 5.

```
(define (operator d-rast)
    :parameters (?Z ?X ?Y)
    :precondition (and (selected ?Z ?X) (data ?Y) )
    :effect (and
        (forall (?A ?B)
            (when (displayed-in ?Z ?X ?A ?B)
                (not (displayed-in ?Z ?X ?A ?B))))
        (forall (?F ?D ?I ?I2)
            (when (and (contains ?Z ?X ?F ?I) (displayed-in ?F ?I ?D ?I2))
            (not (displayed-in ?F ?I ?D ?I2)) ))
            (when (exists (background-colour ?Z ?X ?C))
                (not (background-colour ?Z ?X ?C)))
        (forall (?F1 ?B1 ?I3)
            (when (and (contains ?Z ?X ?F1 ?I3)
                (background-colour ?F1 ?I3 ?B1))
                (not (background-colour ?F1 ?I3 ?B1)) ))
        (displayed-in ?Z ?X map ?Y) )
    )
```

"The effect of displaying some raster data is that the currently selected window now has that map present in it. This map overwrites both the background colour and the previous contents of the window, including that of any frames contained within the window".

Figure 5: A planning representation of a GIS command, also expressed in terms of natural language
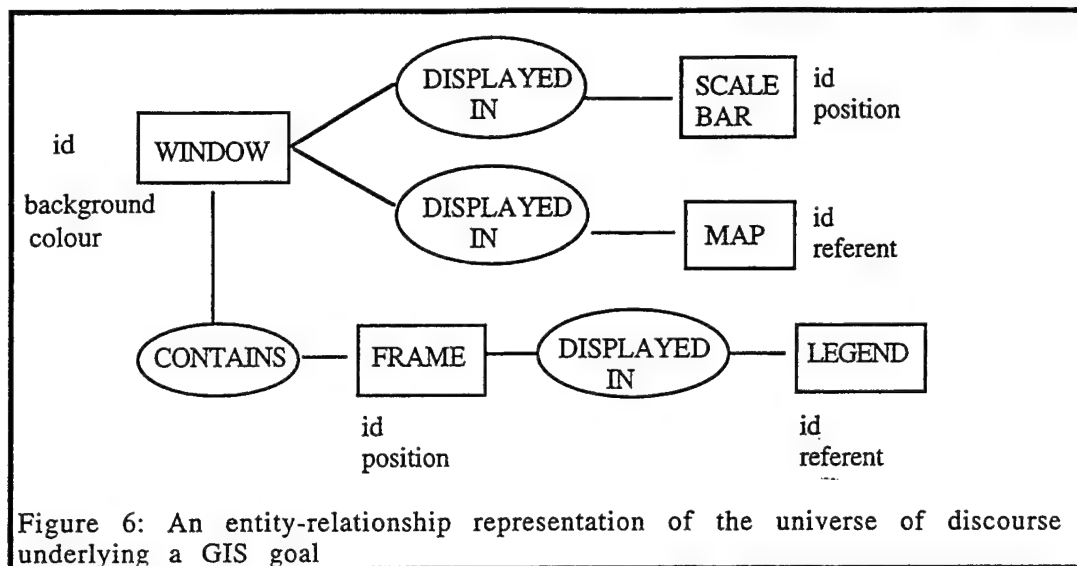
The main feature of this example is its support of universal quantification over a dynamic object universe. Objects here refer to somewhat tangible entities, such as data files, and also to more ephemeral things such as the contents of graphics windows.

A representation of the seven-step plan of Figure 2 is returned, in the best case, in 0.8 secs on a Silicon Graphics 200 MHz MIPS machine running GNU Common Lisp 2.1, and in 1.9 secs on a standard Sun Sparc2 running Lucid Lisp 4.0. This performance, although comfortable, is less impressive than has previously been reported. Possible reasons are that this GIS domain is more demanding in terms of (a) complexity of the action descriptions, and (b) average length of the plans. As support, others have nominated plan lengths of 10 steps as being extraordinary, and have stressed the necessity of domain-dependent search heuristics (Chien, 1994). Unfortunately, this strategy generally militates against soundness and completeness. The current experience also suggests that goal order has a significant effect upon performance (swamping platform differences), which provokes the issue of whether some parser could optimise this order, preferably in a domain-independent fashion. In fact, those planners which infer a hierarchy at plan-time (Barrett & Weld, 1994), as opposed to relying upon a store of skeletal plans (Chien, 1994), may be seen as addressing this issue.
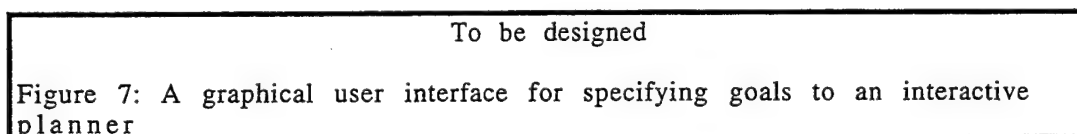
Two interfaces require attention before this investigation of feasibility may progress. The first is between the planner and the application. It is routine to transform the output of the planner into a script which may be submitted to the operating system. (However, if one also envisages that the planner may respond to the application, then additional work is required, as discussed in section 4). The main interface concern at this point is that with the user. Clearly, after criticising contemporary GIS user interfaces, it would be inconsistent to claim that the predicate logic interface of Figure 4 represents an advance in useability! In its raw form, this interface poses a number of problems:

. Lisp/UCPOP syntax

. the semantics of predicate calculus, including conjunction, negation, and existential & universal quantification

. lack of guidance about the types of goal statements which are possible

These problems are also familiar from the database world which, once again, is no accident, given planning's intimate relationship with logic. This recognition has the advantage of providing a certain amount of conceptual leverage; for example, it allows one to compare and contrast goal description languages (and techniques) with more familiar database query strategies, despite the fact that plan synthesis is not generally regarded as an information retrieval task. The predicate logic interface of Figure 4 may be seen as an analogue of SQL: declarative (in comparison to its predecessors), demanding (for inexperienced users), and also limited by its first-order formalism (eg, it is difficult to pose a meta-query about which predicates are available, because that entails treating predicates as variables). The universe of discourse of this domain may be also represented in terms of an entity-relationship diagram, as shown in Figure 6.

Figure 6: An entity-relationship representation of the universe of discourse underlying a GIS goal
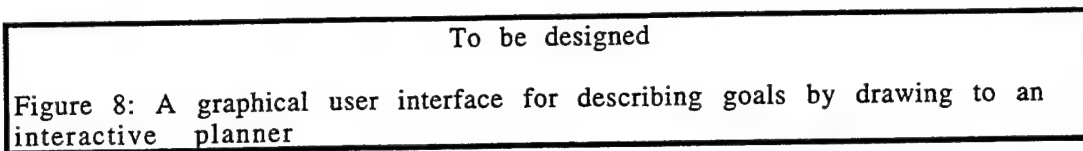
One advantage of the diagramming of Figure 6 is that a certain ontological structure is revealed in comparison to untyped predicate logic, eg, some predicates function as attributes of entities (position, background-colour) whereas others serve to relate two entities (contains, displayed-in). It is also notable that one entity (window) is not present in the natural language goal specification of Figure 4; ie, this entity is consequential upon the goal of displaying maps. It would seem important to impress these distinctions upon end-users, and a graphical interface naturally suggests itself. A graphical interface would be expected to have the additional advantage of eliminating problems of syntax, in the conventional fashion. An example of this approach is shown in Figure 7.

To be designed

Figure 7: A graphical user interface for specifying goals to an interactive planner

Employing a similar graphical interface for their softbot, (Etzioni & Weld, 1994) suggest that such an approach reduces users' discomfort with logic. More precisely, such an approach may be expected to reduce problems of syntax, but the ability of graphics to facilitate a grasp of the semantics of logic is considered in this paper to remain an empirical question. An extension of this approach is also suggested by the insight that a number of the predicates (position, contains, displayed-in), by virtue of their spatial associations, lend themselves fairly readily to a graphical, as opposed to linguistic, style of definition. It is possible to envisage users being presented with a palette of domain entities, similar to an interactive drawing package. These entities are instantiated by an act of selection, and their properties and relationships are

9

defined by drawing actions wherever possible. An example of this approach is shown in Figure 8.

To be designed
Figure 8: A graphical user interface for describing goals by drawing to an interactive planner

At this point, the reader will observe that we have come full circle. Conventional graphical GIS interfaces were criticised because of their imperative nature, and pessimism was expressed about the possibility of expressing all operations in terms of direct manipulation. However, the interface of Figure 8 is obviously heavily influenced by direct manipulation ideals, with the main difference being that it represents an abstract 'sketch' of a conjunction of goals (which some planner is subsequently expected to fulfil) rather than an arrangement of domain entities which could be satisfied by a single underlying command. (In practice, true, 'direct' manipulation of domain entities is an impossibility, as all computer graphics are necessarily abstract representations of something else to a greater or lesser degree. Figure 8 is a representation of Figure 1, which in turn is a representation of a real-world road network). Further experience is required to resolve these issues but, in the interim, it may be speculated that the interface of Figure 8 provides a synthesis between conventional, object-oriented graphics and newer, AI-derived techniques.

4. Planning Embedded within User Interface Management Systems

UIMS provide a structured framework within which interactive software may be developed. This section of the paper attempts to draw some implications for interactive planners which may be embedded within such frameworks.

The specification of appropriate architectures for UIMS is controversial and, although this has not been articulated clearly in the literature, two methodological 'camps' may be discerned. The original philosophy was that a UIMS was a means of 'front-ending' some existing application, typically in a bid to upgrade its interface, eg, (Green, 1985). More recently, attention has turned to the issue of interfaces for contemporary, object-oriented applications, which is considered by some to militate against the notion of a separate front-end in the classical sense (Paton et al, 1994). Regardless of this controversy, UIMS have a firm heritage in the CASE-like notion of executable specifications and model-based approaches to

software development. More particularly, the original UIMS approach at least involves the front-end containing an executable model of the capabilities and limitations of the application to which it is interfaced. A variety of roles have been proposed for this model (Alty & McKell, 1986; Olsen, 1987) including the trapping of semantic errors on the part of the user, responding to queries about the consequences of application commands, and even the provision of tutoring facilities.

This model typically employs a planning-like representation of application actions, preconditions and effects, but more for the purpose of deriving states from given action sequences, rather than the converse, eg, (Hudson & King, 1986; Hurley & Sibert, 1989; Sukaviriya et al, 1993). Traditional UIMS assume that the main purpose of the interface is to enable users to invoke application callbacks, and dynamic models of application state are maintained in order to control the state of interface widgets. Whilst this forward-reasoning task is computationally unremarkable, it fulfils a pragmatic requirement which embedded, interactive planners need to address. There are isolated instances of comparatively unsophisticated planners being employed to answer "How can I ...?" queries from the user (Senay et al, 1990). In this case, the UIMS experience suggests that planners cannot simply execute automatically the first plan which they synthesise; explanation facilities involving plan alternatives would seem warranted as an option.

A UIMS typically responds to and possibly filters the output of the application and so, by extension, should an ideal interactive planner. This requirement potentially strains the capabilities of current software agents if the application output includes error diagnostics. Most basically, the planner needs to parse that output in addition to its normal task of parsing conjunctions of predicates entered by the user. More challengingly, it may be envisaged that those diagnostics are received because of some relaxation of the principle that exogenous state updates do not occur, eg, because a second user has deleted some file in between the time of planning and the time of execution. In that case, the demands are starting to resemble those of real-world robotics domains, and error recovery and re-planning become an issue,

UIMS provide a valuable conceptual insight: that of planning-like representations forming the basis of executable meta-models, or schemas, of applications. Unlike database schemas, these are operator-centred rather than data-centred, and are also more proactive; ie, these mediate all user interaction rather than being

consulted passively as ancillary sub-systems. It is notable that the granularity of this schema is largely driven by end-user concerns, ie, it is a function of the granularity of the anticipated goals which might be posed to the system. This raises the intriguing possibility that the 'application' need not be some singular system such as a GIS or the Unix operating system, and could be considerably larger, such as a C3 system. In other words, the daunting requirement to provide an interactive model of every aspect of a C3 system is evaded if the operations visible to end-users are not of a fine level of detail. A more problematic aspect which may be envisaged is the need to model the non-instantaneous effects typical of real-time control systems.

Taking a broad view, developments in planning may be said to indicate a renaissance of the general problem solver approach (Newell & Simon, 1972) to AI. Contemporary planners involve a clear partitioning between domain-independent search algorithms and domain-specific operator modelling. However, even that modelling is bound to conform to a common representation of preconditions and effects. The aspirations towards generality of this approach distinguishes interactive planners from other AI support systems in the spatial information processing field, such as expert systems (Srinivasan & Richards, 1993), despite the fact that some of these do synthesise plans on the basis of their own representations, eg, (Guenther et al, 1993). There are other differences of intent: planners tend to model the given and publicly verifiable semantics of applications, whereas expert systems have tackled the modelling of more abstract domains, such as user tasks and knowledge. On the other hand, planning formalisms have been applied to modelling user knowledge (Blandford & Young, 1993), (Copas & Edmonds, 1984). It is an open question whether planners should be regarded as succeeding or instead supplementing expert systems, where user support is concerned.

5. Conclusion

Planning technology has matured to the point whereby this paper demonstrates that it is feasible to build software agents which perform some significant tasks, such as supporting the users of GIS. A broad view of these developments suggests that more is involved than simply the provision of intelligence: paradigms of user interaction may evolve from an imperative towards a more declarative style. The advent of interactive planners raises design issues of goal description techniques, and some alternatives have been examined within this paper. The relationship of planners with

UIMS technology has also been examined, with the conclusion that embedded planners are subject to increased functional demands, such as reasoning from actions to states, providing explanation facilities to users, and conducting dialogues with applications.

References

Alty, J. L. & McKell, P. (1986). Application modelling in a user interface management system. In M. Harrison & A. Monk (Eds.), People & Computers I, pp. 319-335. Cambridge: Cambridge University Press.

Barrett, A. & Weld, D. S. (1994). Task-decomposition via plan parsing. In Proceedings of ?th National Conference on Artificial Intelligence (AAAI '94), . Morgan Kaufmann?

Blandford, A. & Young, R. M. (1993). Developing runnable user models: separating the problem solving techniques from the domain knowledge. In J. L. Altyet al (Eds.), People and Computers VIII (HCI '93), pp. 111-121. Cambridge: Cambridge University Press.

Chien, S. (1994). Using AI planning techniques to automatically generate image processing procedures. In K. Hammond (Ed.), Proceedings of Second International Conference on Artificial Intelligence Planning Systems, pp.219-224 . Chicago: AAAI Press.

Copas, C. V. & Edmonds, E. A. (1984). Executable task analysis: integration issues. In People and Computers IX (HCI '94), Cambridge: Cambridge University Press.

Egenhofer, M. J. & Richards, J. R. (1993). Exploratory access to geographic data based on the map-overlay metaphor. Journal of Visual Languages and Computing, 4, 105-125.

Etzioni, O. (1993). Intelligence without robots (a reply to Brooks). AI Magazine, 14(4).

Etzioni, O.et al (1994). Tractable closed world reasoning with updates. In Proceedings of 5th International Conference on Principles of Knowledge Representation and Reasoning (KR '94), .

Etzioni, O. & Weld, D. (1994). A softbot-based interface to the Internet. Communications of the ACM(July).

Fischer, G. (1991). The importance of models in making complex systems comprehensible. In M. J. Tauber & D. Ackermann (Eds.), Mental Models and Human-Computer Interaction 2, Oxford: North-Holland.

Green, M. (1985). The University of Alberta User Interface Management System. Computer Graphics, 19(3), 205-213.

Guenther, O.et al (1993). RESEDA: a knowledge-based advisory system for remote sensing. Applied Intelligence: the International Journal of Artificial Intelligence, Neural Network and Complex Problem-Solving Technology, 3(4), 317-341.

Hudson, S. E. & King, R. A. (1986). A generator of direct manipulation office systems. ACM Transactions on Office Information Systems, 4(2), 132-162.

Hurley, W. D. & Sibert, J. L. (1989). Modeling user interface-application interactions. IEEE Software(January), 71-77.

Newell, A. & Simon, H. A. (1972). Human Problem Solving. Englewood-Cliffs, USA: Prentice-Hall.

Olsen, D. R. (1987). Larger issues in user interface management. Computer Graphics, 21(2), 134-137.

Paton, N. W.et al (1994). Integrated architectures for database interface development. IEE Proceedings on Computers and Digital Technology, 141(2), 73-78.

Penberthy, J. & Weld, D. (1992). UCPOP: a sound, complete, partial order planner for ADL. In Proceedings of 3rd International Conference on Principles of Knowledge Representation and Reasoning, pp.103-114 .

Scopigno, R.et al (1990). Visual language approach: a proposal for improving user friendliness and effective use of GIS. In Proceedings of European Conference on Geographical Information Systems (EGIS '90), pp.1024-1034 . Amsterdam: EGIS Foundation.

Senay, H.et al (1990). Planning for automatic help generation. In G. Cockton (Eds.), Engineering for Human-Computer Interaction, pp. 293-311. Amsterdam: Elsevier Science.

Srinivasan, A. & Richards, J. A. (1993). Analysis of GIS spatial data using knowledge-based methods. International Journal of Geographical Information Systems, 7(6), 479-500.

Sukaviriya, P. N.et al (1993). A second generation user interface design environment: the model and the runtime architecture. In S. e. a. Ashlund (Ed.), Proceedings of Interchi '93, pp.375-382 . Amsterdam: ACM Press.

Weld, D. S. (1994). An introduction to least commitment planning. AI Magazine, 15(4), ??

Strategic C3I Systems as Situated Action

John O'Neill

Information Technology Division
Defence Science and Technology Organisation, Department of Defence
PO Box 1500, Salisbury, South Australia, 5108
Email: John.ONeill@dsto.defence.gov.au

Abstract: The nature of strategic C3I systems is fundamentally different to operational and tactical C3I systems. This paper proposes that strategic C3I systems are problem formulation systems not problem solving systems. Problem formulation systems define situation representations for operational and tactical C3I systems to solve. A situated action model is used that enables the perceptual, cognitive and action representations to mutually adapt to meet the needs of the individuals and situation. A framework is described for situated reasoning systems that enables the construction of situation representations by a group of domain experts. A key finding from this approach is that interoperability is defined as the ability to mutually adapt representations between strategic, operational, and tactical CSS and intelligence systems. This research is currently being applied to the Directorate of Joint Planning at HQADF to support the strategic planning process.

1. Introduction

Whilst development of C3I systems at the tactical and operational levels have met with some success, developing strategic C3I systems has proven more difficult. It is now recognised that articulating the requirements of a strategic C3I system *a priori* is not completely possible. Instead, the requirements are a function of the needs of an individual acting in a situation [1, 2].

The nature of strategic C3I systems is fundamentally different to operational and tactical C3I systems. This paper proposes that strategic C3I systems are *problem formulation systems not problem solving systems*[*]. It will be argued that traditional "black box" symbolic processing approaches are inadequate for problem formulation and that a situated action perspective is required. A situated action model for strategic C3I systems is proposed. Based on this model, a framework for developing situated action reasoning systems is described.

2. Symbolic Processing versus Situated Action

An increasing number of research areas are encountering the dichotomy between the objectivist, or internal cognition, viewpoint and the role of the environment. In Cognitive Science this is characterised as symbolic processing versus situated action [3], Organisation Theory views the debate between closed, rational and open, natural systems [4], Decision-Making theory compares rational and naturalistic decision-making methods [5], Cognitive Psychology compares the role of semantic and episodic memories [6], Artificial Intelligence compares symbolic systems with connectionist approaches [7], research into Conceptual Structures compares similarity-based and theory-driven categorisation [8], Problem Solving compares routine and novel problems [9].

The internal cognition, or symbolic processing, approach relies on an internal or "black box" representation of the world for solving problems [10]. Data is input from the environment, internal context-free reasoning processes manipulate the data and select an appropriate action to act on the environment.

In contrast, the situated action or social constructivist model emphasises the role of the environment, the context, the social and cultural setting, and the situations in which individuals find themselves [3]. The situated action approach has three components: perceptual

---

[*] This paper is not arguing that only strategic C3I systems formulate problems. On the contrary, all C3I systems require the ability to formulate problems. However, strategic C3I systems have a higher percentage of novel problems requiring problem formulation, hence their emphasis in this paper.

component, cognitive component and action component. The perceptual component includes the representations used to input data from the real-world. The representations chosen for the perceptual component influences what is "seen". The cognitive component consists of the representations to interpret this real-world information, perform planning operations, and derive action plans. The action component involves the representations required to actually perform the tasks produced by the cognitive component. The situated action approach recognises that the perception representations, cognitive representations, and action representations are mutually dependent. The cognitive representations are dependent on perceptions and actions. Data perceived changes the cognitive representations. Cognition changes the actions, and the way the world is perceived. Using this paradigm, situated action tightly integrates the symbolic and connectionist approaches.

Within the symbolic processing community, there is growing awareness of the role of context in reasoning [11, 12], natural language [13], and vision. Polya [14] argues that the ability to reformulate problems is one of the keys to learning mathematics. Simon [10, 15] argues that the ability to change representations is an inherent feature of the symbolic approach based on his research into cognitive psychology and Artificial Intelligence. However, the ability to formulate and reformulate representations is a neglected area of Artificial Intelligence, which continues to focus on problem-solving. Situated action approaches identify the need for problem formulation and integrates the symbolic and connectionist approaches. The following section argues that strategic C3I systems require the ability provided by the situated action model to formulate and reformulate representations.

3. The Nature of Strategic C3I Systems

This section explores C3I systems that support strategic planing. Previous approaches in conceptualising strategic C3I systems will be analysed. This analysis reveals the deficiencies in viewing strategic C3I systems as problem-solvers. A comparison is then made between symbolic processing and situated action approaches for conceptualising strategic C3I systems.

3.1 Strategic Planning

Strategic planning in the Australian Defence Force (ADF) is performed at HQADF. There are two types of strategic planning: deliberate and immediate planning. Deliberate planning is longer-term planning that aims to predict possible future threats to Australia's national interests. The output of the deliberate planning process is a set of contingency plans to counter these threats. Immediate planning is short-term planning that is reactive to a crisis situation. Where possible, immediate planners adapt contingency plans to meet the needs of the situation. However, the nature of crisis situations means that some crises cannot be predicted. In these situations, the immediate planners must perform the deliberate planning process in compressed timescales.

Military strategic planning aims to integrate the military response with the national response to achieve the national end-state. The strategic plan documents the situation, background information, government guidance, the military end-state, the strategic concept of operations, and the resources and constraints for achieving the military end-state.

3.1.1 The Strategic Planning Process

There are five steps to the strategic planning process: strategic intelligence, military threat analysis, government guidance, ADF response, and production of the strategic plan. The intelligence community is responsible for the strategic intelligence which documents the capabilities, intentions and events of interest of other countries in the region of interest. Strategic intelligence covers political, economic, military, diplomatic and legal factors.

Strategic planners use the military threat analysis, in collaboration with the intelligence community, the government, and other government departments, to determine possible

courses of action by an adversary and decide whether any of these courses of action may threaten Australia's interests. The courses of action analysis involves determining what types of events may escalate into potential conflict situations. Underlying this analysis is a study of the centre of gravity for an adversary investigating the basis of an adversary's power structure.

The government guidance generically defines Australia's national interests, and specifically defines Australia's national end-state for a given situation. The ADF response investigates how Australia may defeat an adversary to achieve the national end-state. The ADF response relies on using the centre of gravity analysis and knowledge of the ADF's capability and preparedness to defeat an adversary. The output of the ADF response is a set of options. The government decides which option is the most appropriate given the political, diplomatic and economic responses that are being developed in parallel. The selected option then forms the basis for developing the strategic plan.

3.1.2 Discussion

The four steps of strategic intelligence, military threat analysis, government guidance and ADF response are *not* performed in a serial fashion. Instead, they are conducted in parallel with issues raised in one step leading to further investigation in other steps. For example, investigating the military threat analysis may reveal questions about the adversary's capabilities that require further work by the intelligence community.

Each of the four steps of strategic intelligence, military threat analysis, government guidance and ADF response use different representations to support their different types of analyses. The strategic plan is a fifth representation that communicates to the operational level commander the end-state that needs to be achieved, without communicating all the analysis that derived this end-state and concept of operations. Due to time constraints, the operational level planning is often conducted in parallel with the strategic planning. Parallel planning enables a shorter total planning window and greater operational input to the strategic plan. The major disadvantage of parallel planning is that the operational planning process must be adaptive to the continual changing requirements from the strategic level as new information is sought, and the various analyses developed.

Strategic planning is not performed in isolation. The strategic planning process is performed by a core team of experts who "invite" other experts to participate depending on the nature of the situation and the type of analysis being performed. The strategic planning process is linked into the political and diplomatic planning processes by committee meetings at the one-star and two-star levels. These committee meetings aim to develop a "ladder of escalation" that integrates the military, diplomatic and political responses.

3.2 Previous Approaches

Previous approaches for developing strategic C3I systems have viewed these systems as extensions of operational- and tactical-level problem-solving. Tactical C3I systems solve problems at the unit level, operational C3I systems solve problems at the force-level, strategic C3I systems solve problems at the joint- and coalition-force-level. The information requirements for a strategic C3I system can then be viewed as a superset of the operational and tactical C3I systems. To cope with information overload, it has been assumed that tactical and operational information can be summarised, and key performance indicators identified, to meet the strategic commanders requirements.

Defining this set of summarised information and key performance indicators has proven problematic for four reasons. Firstly, the strategic environment is characterised by equivocality, uncertainty, and inconsistency [16] which constantly changes the underlying information requirements. Secondly, strategic commanders don't make decisions based solely on summarised strategic information and key performance indicators. Sometimes they require specific, detailed tactical information. The tactical information required will depend on the

needs of the situation and of the individual commander. The third problem is staff turnover. Even if the information requirements of an individual commander could be completely defined, current military policy rotates officers on a two to three year cycle. Staff turnover results in constant redefinition of the information requirements for a strategic C3I system. Fourthly, strategic commanders rely heavily on face-to-face formal and informal meetings for collecting information. The information collected in these meetings consists of both intangibles, such as psychological assessments and morale, and tangible situational and organisational information.

3.3 Symbolic Processing versus Situated Action Approaches to Strategic C3I Systems

The information requirements for strategic C3I systems appear to be huge and unable to be completely defined. Yet research into cognitive psychology has shown that humans cannot cognitively cope with these volumes of information. It is interesting to investigate why such emphasis has been placed on totally defining the information requirements of strategic C3I systems. Pre-defining these information requirements assumes using the symbolic processing model and views strategic C3I systems as *problem-solvers*. Problem-solving systems need to navigate through information- and task-spaces to produce solutions. Clearly, this approach will produce less than optimal solutions if the information-space cannot be completely articulated.

The situated action model views strategic C3I systems as *problem formulation* systems. Problem formulation systems define problem representations for operational and tactical C3I systems to solve. In the situated action approach, strategic C3I systems formulate, or reformulate, cognitive, perceptual and action representations. These representations are developed as a result of interpreting and understanding real-world events, and formulating a situation. The information requirements, as expressed in these representations, are tailored to the needs of the individuals and situation.

The development of situation-specific cognitive, perceptual and action representations appears to fit the cognitive information requirements of expert decision-makers as defined by Klein [5]. In comparison, the symbolic processing model fits the cognitive information requirements of novice problem solvers.

3.4 Summary

Strategic planing defines the military response required to achieve the national end-state. The five steps of the strategic planning process are: strategic intelligence, military threat analysis, government guidance, ADF response, and production of the strategic plan. These steps are performed in parallel, with one step deriving new requirements that the other steps must satisfy in order to produce a coherent plan. The strategic planning process is often performed in parallel with the operational planning process in order to satisfy planning deadlines. The strategic plan records the results from the various analyses conducted during the strategic planning process. Strategic planning is a group process, with experts invited to strategic planning sessions as required.

Pre-defining information requirements for strategic C3I systems has proven problematic due to: environmental factors, individual cognitive information requirements, frequent staff turnover, and the need for informal and informal information flows. Symbolic processing approaches for developing strategic C3I systems have proven inadequate due to the inability to pre-define an information space for problem solving. A situated action approach views strategic C3I systems as problem formulation systems. The next section develops a situated action model for strategic C3I systems.

4. A Situated Action Model for Strategic C3I Systems

A situated action model for viewing strategic C3I systems as problem formulation systems is developed in this section. This model is then applied to a military organisation, and defines the responsibilities and interactions required between organisation units.

4.1 The Situated Action Model

The situated action model views strategic C3I systems as undergoing continual adaptation through the process of problem formulation and reformulation. There are six components to this model as shown in Figure 1: perception component, problem formulation component, cognitive component, action component, problem reformulation component, and historical learning component. These components will be presented sequentially, however, they are inter-meshed and are constantly being revisited.
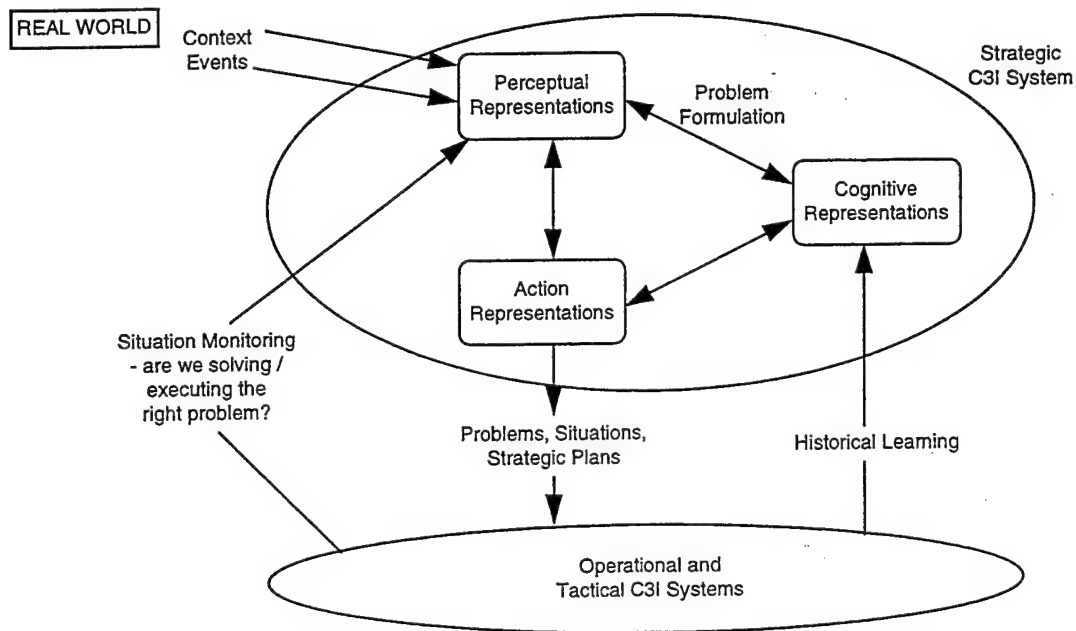

Figure 1. The Situated Action Model

4.1.1 Perceptual Component

Real-world events are perceived by the systems perceptual component. These perceptions are then interpreted. Do the perceive events relate to a current situation, or are they something else? If they're something else, perform further processing using contextual information to understand the nature of these events. Will these events cause us a problem? If so, start the problem formulation component, if not, file or discard the information.

The perceptual component for a strategic planning process includes the sources of information. These sources include the strategic planners, and their networks of experts. It also includes the strategic intelligence and government guidance information supplied by the intelligence community, and other government departments respectively.

4.1.2 Problem Formulation Component

The problem formulation component aims to formulate a problem description and appropriate cognitive, perceptual and action representations. This is achieved by firstly collating information about events and contextual information, including different

perspectives of this information. These perspectives are then interpreted to understand the nature of the situation. This understanding is then used to develop a description of the problem representation, and appropriate cognitive, perceptual and action representations.

Determining what the problem is can be considered the basis of strategic planning. The military threat analysis step of the strategic planning process is the "melting pot" where information from many sources is analysed to determine an adversary's possible courses of action, derive threats to Australia's national interests, and thus define the problem situation.

4.1.3 Cognitive Component

The cognitive component is used to reason about the world and perform strategic planning. The problem formulation and cognitive reasoning processes develop constraints on possible actions, for example, formulating a situation as a UN peacekeeping operation precludes the use of pre-emptive air strikes. These processes select and redefine relevant concepts for perception and reasoning, produce goal statements, identify force structures, appoint commanders and identify command and control arrangements. A result of the cognitive reasoning process is that the perceptual, cognitive and action representations may need to be reformulated.

The cognitive component of the strategic planning process is the analysis of the ADF's response. This analysis is tightly coupled with the military threat analysis that drives the problem formulation component since both analyses require use of the centre of gravity analysis.

4.1.4 Action Component

The output of the strategic C3I process are actions, orders, problem situations and strategic plans. Problem situations and strategic plans define the know-what. Know-what consists of the problem definition, identifies the constraints, and command and control arrangements, and defines the relevant information requirements at a strategic level, including the specific tactical information required. These problem situations and strategic plans are used by the operational and tactical C3I systems to plan how to resolve the problem situation, and execute these plans.

The action component of the strategic planning process is the strategic plan.

4.1.5 Problem Reformulation Component

Strategic C3I systems are not just problem formulation systems. They are also problem reformulation systems when required. Strategic C3I systems perform situation monitoring of the operational and tactical planning and execution to ensure that the right problem is being solved. If real-world perceived events don't fit the cognitive representations for a situation, problem reformulation may be required. An example situation where problem reformulation was required was Operation Restore Hope in Somalia which started as a humanitarian relief mission and ended as a law enforcement mission.

4.1.6 Historical Learning Component

Once a situation is resolved, the historical learning component is used to evaluate the effectiveness of the representations developed and processes employed. One outcome of historical learning may be that representations are reformulated to increase their likelihood of being re-used in future situations.

4.2. Assigning Organisations to the Situated Action Model

This section explores which organisations would be responsible for the components identified in the situated action model for strategic C3I systems. The implications for systems developed by these organisations is discussed.

4.2.1 Assigning Organisations

The cognitive representations would be produced by either the deliberate or immediate joint planning group at a strategic headquarters, depending upon the requirements of the situation.

Problem formulation may be produced by the joint planners at a strategic headquarters, the intelligence community, or by government direction.

The perceptual representations have two sources. The intelligence community create perceptual (and cognitive) representations of the real-world based on an understanding of external agents / organisations / governments. Secondly, each organisational role, or area of expertise within the organisation, creates a perceptual representation of the real-world based on their domain knowledge. A major component of strategic C3I systems is integrating these different perspectives of the real-world providing a richer basis for interpreting and understanding the world and formulating problems.

4.2.2 Implications

Interoperability based on pre-defined information flows and conceptual models are inadequate. Situated action requires the definition of new representations to meet the needs of a situation. These representations need to propagate across all C3I systems in a process of *mutual adaptation*. Thus, the information flows and conceptual model requirements are produced as a result of the problem formulation process for each situation. Interoperability is the ability for C3I systems to mutually adapt their representations.

Command Support Systems (CSS) and Intelligence systems must adapt to situations inline with each other. The situated action model specifies that C3I systems are continually evolving and that the perception representation changes the cognitive representation and vice versa. Therefore, the CSS and Intelligence systems will *require continual adaptations* to each other to meet the needs of a situation.

Strategic C3I systems structure operational and tactical C3I systems. The strategic C3I systems will specify the conceptual requirements (know-what) for each situation for operational C3I systems. Operational C3I systems will specify the conceptual requirements (know-what and know-how at a force-level) for each situation for tactical C3I systems. Tactical C3I systems will specify the conceptual requirements (know-what and know-how at a unit level) for mission planning systems. Current CSS being developed at the environmental and strategic headquarters need to recognise that the nature of C3I systems requires interoperation and mutual adaptation.

4.3 Summary

A situated action model for strategic C3I systems has six components: perception component, problem formulation component, cognitive component, action component, problem reformulation component, and historical learning component. These components are spread throughout the military community encompassing intelligence, strategic, operational and tactical CSS systems. The requirements for interoperability amongst these systems requires not only interoperability at an information level, but mutual adaptability at a representation level.

5. Building Situated Action Reasoning Systems

Strategic work is viewed as a group of domain experts at a strategic headquarters interpreting events and formulating problem situations. Each expert contributes both domain and organisational knowledge. This approach is similar to Lenat's [17] BEINGs research, where each BEING modelled a domain expert collaborating to solve problems. Collaboration is facilitated by each BEING instantiating a common frame structure.

The framework presented in this section assumes that each expert will employ a different representation and that the experts collaborate to formulate a problem representation and description. The characteristics of domain experts requiring information systems support are explored. A representation is proposed that views concepts as rich objects with multiple definitions, structures and behaviours stored in an organisational ontology, or corporate memory. These concepts are accessed by individuals by "lifting" the appropriate conceptual structure and behaviour from the organisational ontology into a problem-solving representation. These problem-solving representations are then used by individuals performing roles in groups to formulate problems, or situations.

5.1 Characteristics of Experts

The role of a headquarters at any level in an organisation is to integrate the perspectives of different types of work performed and allocate work to accomplish the purpose and goals of the headquarters. Domain experts from each of the different areas of work collaborate within headquarters to integrate their perspectives. Characteristics of these domain experts include:

- different domains of expertise

- each domain has its own ontology and problem-solving methods, requiring different ways of thinking about situations

- these ontologies may be inconsistent across domains

5.2 A Framework for Situated Action Reasoning Systems

The framework shown in Figure 2 provides an environment for formulating problems, creating a new representation for a situation. The central part of this framework is the organisational ontology which views concepts as rich objects. A concept's structure and behaviour are selected and defined for a situation by an individual, often working in a group. These concepts are "stored" in the ontology representation and are "lifted" into problem-solving specific representations. Problem formulation can be viewed as constructing a representation, lifting concepts and, sometimes, integrating different perspectives to create a new structure and behaviour for a concept. This framework assumes that knowledge bases and databases are networked in a distributed object environment.
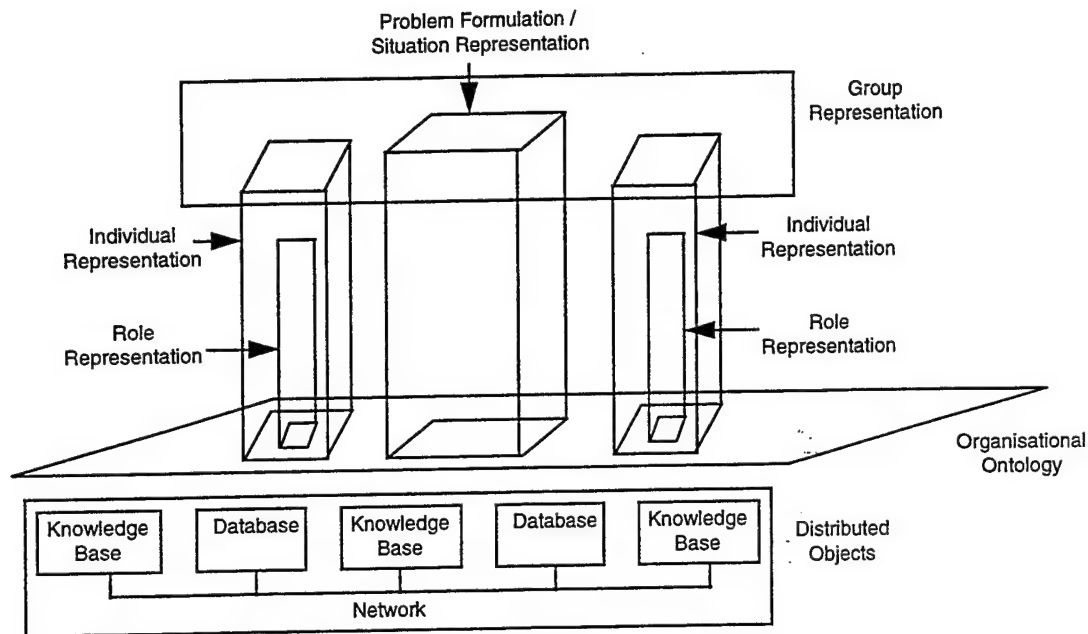
Figure 2. A Framework for Situated Action Reasoning Systems

5.2.1 Concepts as Rich Objects

The problem formulation and reformulation approach of situated action implies that concepts are rich objects. A rich object is an object that can never be completely defined [18] and may have multiple, possibly inconsistent, domain-dependent definitions, structures and behaviours. Defining concepts as rich objects is a new approach in the Conceptual Structures [8] and Linguistics [19] literature and extends the microtheory [20] and contextual reasoning [21] research in Artificial Intelligence.

Viewing concepts as rich objects requires the ability to create and manage multiple representations for a concept. Each of these conceptual representations is context-dependent and specifies a definition, structure and behaviour for the concept.

5.2.2 Organisational Ontology

The organisational ontology stores all concepts and defines an inter-lingua for mapping across concept representations. The inter-lingua requires a richer representation language than the customised, problem-specific representations. For example, CYC is currently employing a second-order logic representation for its inter-lingua, or common-sense knowledge-base. "Lifting" or "bridging" rules [20, 21] are used to lift a concept representation (a definition, structure, and behaviour) from the inter-lingua into the problem-specific representation (for example, frames, semantic networks, first order logic etc) for problem-solving.

Creating multiple representations for a concept requires a richer ontology than simply storing concepts. The ontology must also store the perceptual, cognition and action representations that utilise these concepts. Each conceptual representation requires storing links to the creator (either individual, group or role) and the situation(s) where the representation is used. This provides the contextual knowledge for each conceptual representation. It facilitates historical learning on both an individual and organisational basis, enabling reuse.

23

5.2.3 Individual and Role Ontologies

Role ontologies are organisational structures that define the concepts and workflows for routine problems. Individuals perform roles in an organisational context, and incorporate the role ontology into their individual ontology. In novel situations, individuals extend the role concepts and workflows to meet the needs of the situation. Extending these concepts may require the individual to draw-on previous experiences performing other roles. Therefore, the individual ontology requires support for integrating role ontologies, reformulate concept definitions, and integrating conceptual knowledge across experiences.

5.2.4 Reasoning about Organisational Knowledge using Metonymy

Organisations design structures, create organisation units, allocate roles and assign individuals to roles in order to achieve the organisation's purpose or strategic vision. Section 5 describes how strategic work is performed by a group of domain experts. Each expert contributes both a domain perspective and an organisational perspective to the strategic work process.

Metonymy is proposed as a method for supporting the expert's organisational knowledge. Metonymy is defined as a part standing for a whole, in this case a domain expert is representing an area of organisational knowledge. Organisational knowledge includes knowing who to go to in the organisation to get detailed information and interpret events, it also includes representing the wider organisational requirements and viewpoints at group meetings.

Organisational knowledge is required to support informal information flows, and metonymy is proposed as a mechanism for supporting informal collaboration. Domain knowledge is required to support informal information flows using workflow models.

5.2.5 Support for Problem Formulation

The framework proposed for situated action reasoning systems in this section is designed to support domain experts performing strategic work. The domain experts will be responsible for selecting concepts and formulating situation representations. The reasoning system will provide the basis for storing and retrieving representations, and mapping concepts across representations.

The problem formulation process aims to produce a situation representation, which encompasses a set of perception, cognitive, and action representations for the situation. A situation representation consists of a set of constraints and actions. The situation representation is developed by a group of domain experts selecting appropriate conceptual representations for a situation, and determining the required constraints, goals and actions. The situation representation is linked back to the underlying reasoning to enable other users to access additional contextual information as required.

The situation representation is distributed by the situated action reasoning system updating the representations used by the operational and tactical CSS and the intelligence systems. Domain experts at the operational and tactical CSS provide their domain and organisational expertise to interpret the situational requirements and commence their problem solving activities.

5.2.6 Support for Problem Reformulation

The problem reformulation process involves redefining the situation representation due to the perceived events in the real-world being different to those predicted. This may be due to the wrong problem being solved, or a new constraint being introduced. The problem reformulation process involves listing all the current concepts (and their definitions, structures, and behaviours) in the situation representation and evaluating the effects of

introducing the new constraint. The aim of this process is to determine which concepts are valid, which concepts require reformulation, which concepts are invalid, and which new concepts are required.

The situated action reasoning system aids the domain experts by tracking all concepts for a situation representation, listing the specific definitions, structures and behaviours, and providing a vehicle for evaluating the effects of a new constraint. When the problem is reformulated and a new situation representation is produced, the situated action reasoning system updates the representations used by other C3I systems.

6. Conclusions and Future Work

Strategic C3I systems are problem formulation systems, not problem-solving systems. Problem formulation systems define situation representations for operational and tactical C3I systems to solve.

The situated action model is used to view strategic C3I systems as problem formulation systems. In this model the perceptual, cognition and action representations are constantly being reformulated by a process of mutual adaptation.

Strategic, operational, and tactical CSS and intelligence systems require continual mutual adaptation at a representation level in order to implement the situated action model. Interoperability is the ability to mutually adapt representations between C3I systems.

The central component of a framework for situated action reasoning systems is the organisational ontology. The organisational ontology stores the multiple conceptual representations, in terms of definition, structure and behaviour. It links these concepts to the perceptual, cognition and action representations.

This research is currently being applied to the Directorate of Joint Planning at HQADF to support the strategic planning process.

Bibliography

1. Sage, A.P. "Behavioural and Organisational Considerations in the Design of Information Systems and Processes for Planning and Decision Support", *IEEE Transactions on Systems, Man and Cybernetics*, vol 16 (no 9), pg 909-918, 1981.

2. Andriole, S.J. "Next Generation Decision Support Systems Technology" in ed Andriole, S.J., *"High Technology Initiatives in C3I"*, AFCEA International Press, Washington D.C., 1986.

3. Norman, D.A. "Cognition in the Head and in the World: An Introduction to the Special Issue on Situated Action", *Cognitive Science*, 17 (1), pg pg 1-6, 1993.

4. Scott, W.R. *"Organisations: Rational, Natural and Open Systems"*, Prentice-Hall, Englewood Cliffs, New Jersey, 1967.

5. Klein, G.A. "A Recognition-Primed Decision (RPD) Model of Rapid Decision Making" in ed Klein, G.A., Orasanu, J., Calderwood, R. & Zsambok, C.E., *"Decision Making in Action: Models and Methods"*, Ablex Publishing Corporation, Norwood, New Jersey, 1992.

6. Eysenck, M.W. & Keane, M.T. *"Cognitive Psychology: A Student's Handbook"*, Lawrence Erlbaum Associates, Publishers, Hove and London, 1990.

7. Clancey, W.J. "Situated Action: A Neuropsychological Interpretation Response to Vera and Simon", *Cognitive Science*, 17 (1), pg pg87-116, 1993.

8. Medin, D.L. "Concepts and Conceptual Structure", *American Psychologist*, 44 (12), pg pg1469-1481, 1989.

9. Chandrasekaran, B. "Generic Tasks in Knowledge-Based Reasoning: High-Level Building Blocks for Expert System Design", *IEEE Expert*, vol 1 (no 3), pg 23-29, 1986.

10. Vera, A.H. & Simon, H.A. "Situated Action: A Symbolic Interpretation", *Cognitive Science*, 17 (1), pg pg7-48, 1993.

11. Brezillon, P. *"Proceedings of the IJCAI-93 Workshop on Using Knowledge in its Context"*, 1993.

12. Brezillon, P. & Abu-Hakima, S. *"Proceedings of the IJCAI-95 workshop on Modelling Context in Knowledge Representation and Reasoning"*, 1995.

13. Iwanska, L. *"Proceedings of the IJCAI-95 workshop on 'Context in Natural Language Processing'"*, 1995.

14. Polya, G. *"How to Solve It"*, Doubleday Anchor Books, New York, 1957.

15. Simon, H.A. *"Explaining the Ineffable: AI on the Topics of Intuition, Insight and Inspiration"*, Fourteenth International Joint Conference on Artificial Intelligence, Montreal, Morgan Kauffman Publishers, 1995.

16. Daft, R.L. & Lengel, R.H. "Organisational Information Requirements, Media Richness and Structural Design", *Management Science*, 32 (5), pg 554-571, 1986.

17. Lenat, D.B. *"BEINGS: Knowledge as interacting experts"*, Fourth International Joint Conference on Artificial Intelligence, Tbilisi, USSR, 1975.

18. Guha, R.V. & Lenat, D.B. "Language, Representation and Contexts", *Journal of Information Processing*, vol 15 (no 3), pg p340-349, 1992.

19. Lakoff, G. *"Women, Fire and Dangerous Things"*, The University of Chicago Press, Chicago, 1987.

20. Guha, R.V. *"Contexts: A Formalisation and Some Applications"*, PhD Thesis, Stanford University, 1991.

21. McCarthy, J. *"Notes on Formalising Context"*, Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence, 1993.

A Proposal for a System for Analysts to Manage Competing Hypotheses

Ronnie GORI
Information Technology Division
Defence Science and Technology Organisation
PO Box 1500, Salisbury, South Australia
Email: ronnie.gori@dsto.defence.gov.au

Abstract

This paper describes a proposal for a computer based system to support Intelligence analysts in the management of competing hypotheses. The paper describes the basic ideas underlying the proposed system, and details the key issues to be addressed. It also discusses research, in the enabling technologies of artificial intelligence and reasoning under uncertainty, necessary to make the proposed system increasingly intelligent and autonomous.

1. Introduction

The problem of producing high quality and timely Intelligence is inherently difficult. One aspect of the difficulty arises from the inherently poor quality of much of the information received. Such information may be incomplete, ambiguous or incorrect. Reports of events may be received long after they occurred, and out of sequence. Another source of difficulty is that an enemy will be using counterintelligence techniques, and perhaps also practising deception, to create a false picture of their activities. Furthermore, the amount of information received by Intelligence staff can threaten to overwhelm their information systems and tax their analytical ability.

This report proposes the development of a concept demonstrator for a system intended to address some of the above difficulties. When performing analysis, Intelligence staff draw deductions from the information they receive, whilst paying attention to the perceived accuracy and reliability of the information and its source. They use these deductions to support or refute hypotheses they are investigating. The proposed system, called the Hypotheses, Evidence and Deduction Manager (HEDM), is intended to directly support these analytical processes.

The structure of this paper is as follows. Section Two describes the analysis of competing hypotheses technique, the analytical technique which HEDM is intended to support. It explains why this approach could form the basis of a potentially generic analytical support tool. Section Three gives a brief example of the Intelligence problem, highlighting some of the intricacies of the analysis process.

Section Four describes the vision that underlies this proposal. It explains the basic ideas of the HEDM concept, and details the key entities. It discusses the problem of dealing with uncertainty, and the human computer interaction issues that would need to be addressed. It presents one view of a possible mature version of the system, and discusses some of the research issues that would need to be solved for the mature system to be realised.

Section Five contains some proposals for a programme of work to further develop the HEDM concept. Section Six contains the report's conclusions.

2. The Analysis of Competing Hypotheses

This paper proposes the development of a system supporting an essential aspect of the Intelligence analysis process, that is, the management of competing hypotheses. When analysing information to produce Intelligence, one is essentially determining which of numerous alternative hypotheses or explanations best fits the available information[1] [HEU]. The Competing Hypotheses technique provides a systematic technique and structure to the analysis process.

The crux of the technique is simply to evaluate information received in the light of several competing hypotheses[2]. As part of the process, one determines what information is significant. This entails determining if an item of information supports or refutes, or is consistent with or inconsistent with, a particular hypothesis. For supporting or refuting information, a measure of the strength of the relationship can also be recorded.

A matrix is commonly used to record and present the results of analysis. It has hypotheses shown along the horizontal (top) axis, and evidence shown along the vertical axis. With the technique, one can analyse how sensitive hypotheses are to individual pieces of evidence. The technique can also make explicit any information or indicators that one would expect to see given a particular hypothesis.

The procedure is said to be grounded in decision theory and the psychology of judgement. Hypotheses are said to be useful for storing and retrieving information from long term memory. The advantage of examining multiple hypotheses simultaneously, rather than serially, is that it helps a user to examine a wider range of options, reduces the influence of bias, and eliminates evidence with no diagnostic value [HEU]. Because it can be used to develop an audit trail of how a conclusion was derived, the technique introduces an element of rigour into the analysis process.

Two more comments can be made regarding the technique. Firstly, it can provide an organising framework for other analytical techniques, which may either generate candidate hypotheses or provide evidence to support or refute them. Secondly, the technique is generic. It is equally applicable at the tactical, operational and strategic theatres of Intelligence. In fact, the core technique should be equally applicable to, for example, analysis in the Operations or Force Development areas.

---

[1] The alternative approach, termed naturalistic decision making, might also be called the "My Pet Theory" Approach. In this approach, an analyst (or decision maker) tests new information against a single theory, which is strongly adhered to, sometimes in the face of overwhelming contrary evidence.

[2] The evaluation and analysis of information is a core compenent of intelligence analysis.

3. An Example Problem

In this section, a small number of messages are examined, to determine what intelligence can be derived. The aim is to highlight some of the intricacies of the Intelligence analysis process.

3.1 Sightings in Kununurra

Kununurra is a town in the far northeast of Western Australia. For the purpose of this example, it is assumed that a foreign country is engaged in some low level military activity against Australia on Australian soil. The scenario might be typical of some of the recent Australian Defence Force Kangaroo exercises.

Three messages are received within an Intelligence cell. Assume that all three messages refer to the same day.

(a)   A civilian reports that several enemy soldiers were seen around Kununurra during the late afternoon.

(b)   A schoolteacher reports that several unknown men were seen in the vicinity of Kununurra school after 16:00. They may have been carrying weapons.

(c)   A civilian reports that a number of uniformed men were seen outside Kununurra around 17:00.

Note that of the three messages:

- All refer to Kununurra, but have spatial descriptors with varying degrees of vagueness;

- All refer to potentially overlapping times;

- All sources are civilian. The source of message (b), being a schoolteacher, might be regarded as being more reliable than the average civilian, unless it was specifically known that this was not so. This particular teacher, for example, might be known to be an alcoholic or a compulsive liar.

What could an Intelligence analyst do with this information? She could only say with certainty, that these three messages have been received. The messages might all be wrong, or the observers subject to deception; there may have been no incidents within the Kununurra area within the specified time. Alternatively, the messages could be reporting up to three different incidents. If they represented different incidents, they could refer to the same, or different, groups of enemy soldiers. There is also the possibility that there might have been no enemy soldiers around Kununurra at that time; some of these sightings may have been of Australian soldiers.

The sorts of conclusions that can be drawn are often very dependent on the circumstances and beliefs held, at any particular time. If enemy soldiers had been expected in Kununurra, these reports would probably be taken as confirmation. If enemy soldiers had not been expected, then confirmation of the sightings would most likely have been sought.

The Intelligence domain can be seen to be nonmonotonic; new information can invalidate previously held beliefs. If it was later discovered that the reports were factually incorrect, any inferences made based on the flawed information would need to be revised.

4. The Hypotheses, Evidence and Deduction Manager

This section starts by detailing the vision underpinning the proposed development of the HEDM system. It explains the basic system entities, discusses some of the pertinent issues and projects a vision of a possible mature system.

4.1 The Vision and Goals Underlying HEDM

The vision underlying HEDM is to develop intelligent tools and systems to support Intelligence analysts. These systems should assist analysts to combine flawed and incomplete information so as to build and maintain a credible picture of the analyst's domain of interest, and therefore satisfy a commander's information requirements.

The goal of the work is to research and prototype an intelligent information system that can support the analysis of competing hypotheses. The system would make explicit and support the use of the entities and constructs that analysts use. It should also record any intermediate steps and outcomes of the reasoning process.

The system should allow analysts to propose competing hypotheses to see which, in their view, best matches the available data. Users could record deductions and label these as evidence supporting or refuting particular hypotheses or deductions. They could then investigate, explore and manipulate these entities and their relationships.

The proposed system would explicitly support the storage, and recognition, of information that is inconsistent, and assist analysts' attempts to resolve them. It should propagate deductions into the Intelligence databases seamlessly, and also retract them when previously believed information is shown not to be correct. To be effective, the system will need to be integrated into the general Intelligence working environment.

The system might also develop, from supporting individual analysts to supporting cooperating teams of analysts.

One can envision a staged development of HEDM so that it acquires additional intelligence and autonomy at each stage, although always under human supervision. The first stage is the system as information manager, where the system explicitly records and represents information, but all inferencing is done by the analyst. Here, a large part of the system's value would stem from allowing users to visualise and explore its stored information. At the next level, the system could execute automatically some straightforward tasks under human supervision; it might then be termed an *apprentice level* knowledge based system. As the complexity and sophistication of the system develops, so will the difficulty of the tasks that it could execute. It might then be termed a knowledge based *assistant*. At its capabilities increase, the system might behave as a *critic* or *mentor*, where it could observe the problem-solving behaviours and strategies of an analyst, make

suggestions and point out possible errors. With all these models, the analyst must remain in control of the analytical processes.

An ambitious long term aim could be to develop a system analogous to the SRI Core Knowledge System (CKS) for the Intelligence domain [STR87]. CKS was designed as the reasoning system for an autonomous land vehicle operating in an unconstrained outdoor environment. Several aspects of CKS are salient. It was able to store contradictory information, generated by sensors and knowledge sources, resulting in conflicting and incompatible views of the world. It made no attempts to resolve a contradiction until the information was required, and only to the extent required by a task. CKS also had a vocabulary of knowledge, stored as semantic networks, which reflected the multiple levels of knowledge it contained and the tasks it was designed to execute.

However sophisticated HEDM becomes, it can not be a panacea. The quality of Intelligence products will still very much depend on variables such as the quality of the information received, the training and experience of the staff, and the amount of time available to conduct analysis.

These points are expanded in the remainder of this section. They are intended as a starting point for discussion.

4.2 Some Proposals for the HEDM System

The role of Intelligence analysts, and the Intelligence cells to which they belong, is to generate answers to the pressing information requirements of their commander. This is the information the commander needs to fulfil his or her mission. The Intelligence generated must be relevant and accurate, and Commanders must receive it in sufficient time for it to be useful.

When reasoning deductively, analysts can be said to reason in two different modes, data driven and hypotheses driven. In data driven reasoning, one generates inferences and hypotheses from the available data. In hypotheses driven reasoning, one looks for information to help prove or disprove any of the candidate hypotheses. Analysts switch between the two modes effortlessly. After generating some tentative hypotheses, an analyst will look for specific information to support or refute them. New information may trigger the creation of new hypotheses. The current hypotheses may prove inadequate, and an analyst will need to reexamine the data to form new hypotheses. As has been stated, the ability to analyse information with regard to competing hypotheses is a core competency of Intelligence analysts.

The existence of contradictory and inconsistent information, which give rise to conflicting and incompatible views of the state of the world, is a fundamental aspect of the Intelligence domain. This idea is made explicit by the notion of competing hypotheses. Any computer based system supporting Intelligence staff must be able to handle inconsistent information in some sensible way.

4.2.1 Uncertain and Possibilistic Reasoning

Intelligence analysts must also manage probabilities and uncertainties associated with the entities they manipulate. It is worth differentiating between these concepts.

The probability, or possibility, of a proposition defines the confidence with which it can be asserted. Its uncertainty defines the degree (or lack) of precision of the proposition or its relationships with other propositions[3].

Currently, messages are assigned a rating representing their accuracy and their sources are assigned a rating representing their reliability. Analysts also use linguistic terms such as "possible", "probable", "likely" and "certain" to describe the confidence with which they hold their derived assessments. They generally manipulate these measures of possibility and uncertainty implicitly, rather than explicitly.

To be effective, HEDM will have to enable analysts to attach *Measures of Possibility* (MoP) to hypotheses, deductions and messages, and to examine and manipulate them. For these entities, their MoP (at this stage of the work) will be taken to represent both the possibility of it being true, and the uncertainty attached to it. Early versions of HEDM might rely on users to combine these MoPs; later versions should do so automatically, using some mathematically sound approach that produces results comparable to those derived by analysts.

To make HEDM more natural to analysts, it is suggested that it use the same sort of terminology for possibility that is used currently. These linguistically vague terms could then be mapped onto fuzzy membership functions. Because they can represent a range of possibilities, fuzzy membership functions seem well suited to accurately represent (an interpretation of) the meaning of such terms.

4.2.2 The Basic HEDM Entities (A First Cut)

This section discusses the basic entities that might populate the HEDM system. Their purpose, taken collectively, is to represent the conceptual entities and support the conceptual processes, that allow analysts to transfer unreliable information into high quality intelligence. Although these are believed to be the key entities, it will require further study to confirm whether this is so.

At this early stage, no attempt has been made to formally define these entities; standard English definition are implied. The intention is to give a flavour of the types of entities required, and the sorts of relationships that they might sustain.

The purpose of the Intelligence process is to satisfy the Information Requirements of a commander, in as accurate and timely a manner as possible. The commander's requirements are often somewhat abstract; Intelligence staff will refine these into Significant Information Requirements.

Intelligence cells within the land environment (and the land component of the joint environment) receive many types of information. These include imagery, imagery reports and air reconnaissance reports. However, textual messages are the main source of information received. A large proportion of these textual messages will be in free text. This is inevitable, particularly for low level conflicts.

---

[3] My thanks to Arthur Filippidis and Mark Nelson, both of ITD DSTO, for helping me to clarify these concepts.

From the large quantity of information received, any that is deemed significant should be highlighted. This Significant Information (SGI), of text extracted from a message, will form the basis of subsequent deductions. SGI may be generated manually, implying the need for slick 'cut and paste' tools, or by a parser. A message may give rise to one or more pieces of SGI. Each item of SGI extracted from a message should pick up a single MoP consistent with the reliability and accuracy assigned to the source message.

Intelligence cells will inevitably receive messages that are inconsistent with each other. Any SGI derived from these messages will also be inconsistent. In many cases, an apparent inconsistency may be explained by a message being more recent than another. For this reason, messages, SGI and all entities represented in Intelligence databases should have a timestamp attribute, representing the date and time of the event described in the message.4 The attribute would also allow histories to be generated for specific "real world" entities (for example, combat units) from instances of the representational entities.

Deductions can be derived from one or more items of SGI. It is suggested that there be a standard relationship between SGI and deductions, perhaps termed "gives rise to" and its inverse "is deduced from". It is proposed that each instance of the "gives rise to" relationship have a MoP attribute, to signify the strength of the contribution of a SGI to a deduction; these should also have MoPs, signifying their credibility.

The timestamp attribute for deductions would represent the time when it was made, in contrast to message timestamps, which indicate when the reported event occurred. A timestamp for deductions would allow the generation of histories of deductions regarding particular SGI and hypotheses. Deductions should also express the concept of negation. One should be able to assert that "the identity of a unit is not 811/1 Rifle Battalion", at time t1 with a MoP of p1.

The granularity of deductions and SGI are also issues. A major weakness of the Conclusion Model in a previous ITD concept demonstrator, Techniques for Intelligence Processing Systems (TMIPS), was the 'lumpiness' of Unit Conclusions, and the inability to attach different MoPs to different aspects of a Conclusion [GOR94]. A classic example was that one could be certain that some unit was at a particular location, but have little certainty regarding its identity; the TMIPS Conclusion Model could not express these different MoPs regarding different unit attributes, within a single Conclusion.

A proposed solution was to have very simple, or Atomic Conclusions, that could be combined into Conclusions of the required complexity [GOR94]. One might then assert the deductions: "there is a unit at location X, at event time t1 and MoP p1" and "the identity of the unit at location X, at time t1, is Y, with MoP p2" If Deductions were to be structured in a similar way, each individual strand of reasoning could then be pursued separately. The cost may be additional size and complexity in the databases, and additional complexity in the user interface. The relationship binding these separate Deductions would be conjunctive.

---

4 In a database implementation, an item of SGI would "inherit" the timestamp value of its parent message.

Each Hypothesis represents one possible outcome to a chain of reasoning. As such, they may participate in complex relationships with each other. Several hypothesis may represent mutually exclusive conclusions; the system must be able to represent these disjunctive relationships. It seems likely that hypotheses, like deductions, will need also to participate in conjunctive relationships with other hypotheses.

It would be useful to express hypotheses at different levels of abstraction. Analysts could then construct hierarchies of hypotheses, with those at the highest level providing answers to meet the commander's significant information requirements. These highest level hypotheses might be termed Contexts, to highlight their specific role and to make explicit that they provide a context for much of the analysis that will take place. Contexts relating to the same information requirement will be mutually exclusive.

Deductions largely exist to provide support for hypotheses. The links between hypotheses and deductions can be termed Dependencies, and several subtypes of these can readily be identified. Dependencies may directly support or refute hypotheses. They may provide evidence that is consistent with, or inconsistent with, hypotheses. Their MoP would detail the degree of support provided. It should be possible to define semantic relationships for these, and any other dependency types, as well as the other proposed entities.

It is likely that hypotheses will also provide support, positive and negative, to other hypotheses. The dependency relationships between hypotheses may well be the same as those linking deductions and hypotheses. It may also become apparent that analysts will want to use SGI as direct supports for hypotheses, without having deductions as intermediaries.

Hypotheses and Contexts will also have MoPs. These MoPs will be based on the strength of the evidence that support and refute them. For these entities, it may prove preferable to maintain separate figures for the strength of the supporting and refuting evidence.

Although not part of the 'classical' competing hypotheses technique, classes of entities such as Indicators and Expectations fit very snugly into the proposed structure. Indicators have been defined as "information ... which bears on the intention of a particular enemy to adopt or reject a course of action" [CBT81]. For example, increased enemy reconnaissance of an area may indicate that an attack will take place there in the near future. Such information, drawn from knowledge of an enemy, can provide powerful support for deductions and hypotheses. Similarly, the failure to sight a strong indicator may seriously weaken a hypothesis. The absence of any indication of enemy reconnaissance would reduce the strength of any belief of the imminence of an attack.

Expectations are events which are predicted to occur in a given situation in the future. Typically, doctrine might state that the presence of two rifle regiments deployed defensively is likely to indicate the presence of a third regiment, deployed as cover behind the first two. The TMIPS system had a means of entering Expectations, and could scan incoming and existing information for the required data [PRI92]. It also was able to alert a user when an event, that had occurred regularly in the past, did not occur at the anticipated time.

Finally, users will need a simple way to interrogate the network of SGI, deductions, hypotheses and contexts to determine which hypotheses are most strongly supported at any particular time. These hypotheses, together with their MoP, form the set of current beliefs regarding an enemy situation.

4.2.3 *The Human Interface*

The design of HEDM must ensure that the system is centred around the needs of its users. Specifically, this means that:

(a) users have the core role in decision making;

(b) the system needs to support human problem solving;

This first point is redundant for the early versions of a concept demonstrator, where all reasoning will be executed by a user. However, this will increasingly become an issue if HEDM is developed to become more intelligent and autonomous. An analyst must be able to accept, reject or alter system generated conclusions. She or he must also be able to express preferences for partial or interim solutions, and hence control the system's behaviour.

HEDM proposes to support the problem solving behaviour of analysts. It will do so by explicitly maintaining representations of entities that analysts reason with. It will provide tools to visualise these entities and explore their relationships. It will also provide tools to record the intermediate steps and outcomes of the reasoning process.

As the system moves towards a capability for inferencing, the user's expertise will need to be utilised to as large an extent as possible. This would ensure that the system would deliver the best possible results.

The human engineering aspects of the system, which include the cognitive aspects, require careful consideration. Woods, for example, describes a number of automation disasters where human strategies that met the cognitive demands of a task were undermined by new systems [WOO88]. Automating some tasks may have unforseen consequences, for example, if they involve core skills required for other tasks that are not practised elsewhere. A study showed that photographers, who used light meters, not only lost the ability to judge light conditions by eye, but their sensitivity to shades of light also deteriorated [VAU90]. These examples highlight the complexity of the cognitive engineering aspects of the task and system design.

To be useable, the system must be simple and intuitional. From a user's perspective, the system should ideally have no more complexity than a basic word processing system. The system's cognitive engineering, which includes the design of the user interfaces, is critical given the potential quantities of information and the complexity of their relationships.

As an example, messages, SGI, deductions, dependencies and hypotheses all have MoP attributes. In practice, analysts do not seem to pay close attention to every individual MoP in a chain of reasoning. Storing and representing this information explicitly potentially will allow analysts to derive Intelligence that is more soundly based. It is crucial that this information is presented to analysts in a manner that

allows them to extract and add maximum value, rather than be overwhelmed by clutter and detail. Users should also be able to interrogate the system in a way that will control and vary the degree of uncertainty displayed.

Many working analysts believe that data entry is a bottleneck that very much limits the value of any analyst support system. If data entry for HEDM is not to become an issue, the system must impose little or no additional workload on users entering information. The advent of electronic messaging, structured messages and message databases will partially moderate the problem. Because in this domain there will always be messages that are mostly unstructured text, the generation of SGIs from messages will remain a critical issue.

The collection of HEDM entities should serve several purposes. It will provide a record of how hypotheses are derived, making the Intelligence process more transparent to other analysts. It should act as a means of eliciting the workings of the Intelligence process. It should also provide insight as to how meaningful explanations may be structured in a more automated system.

4.2.4 One View of a Mature System

As work on HEDM progresses, the aim would be to progressively add more intelligence to the system, so that it could provide increasing degrees of automated support. This ambition implies the continuing addition of additional knowledge and knowledge sources. Some components of a mature HEDM system appear in Figure 1.

It should not be difficult to justify the individual components of the reasoning engine. An intelligent HEDM would need to understand spatial and temporal concepts. The doctrinal box represents the military knowledge, of both enemy and own forces, that the system would require. Constraint based reasoning deduces values of object attributes by collecting information restricting their possible range. Constraints can be regarded as partial descriptions of entities. They also can be treated as goals to determine if they can be satisfied. This style of reasoning allows a system to pursue a least-commitment style of problem-solving [HAY83].

The underlying reasoning system would probably be provided by a truth maintenance (TMS) or belief revision system. De Kleer explains truth maintenance in the context of problem solving in very large search spaces [DEK86]. TMS systems cache results gained in one part of the search space, and can apply them to other appropriate parts of the space. The alternative would be to recreate the cached results before continuing with the problem solving.

In an Assumption - Based TMS, for example, each deduction is labelled with the sets of assumptions under which it holds [DEK86]. These assumptions include inferences made by the problem solver. There is no necessity for the overall knowledge and data bases to be consistent, as inconsistencies can be represented explicitly in terms of diverging assumptions. It can perform non-monotonic reasoning, because an assumption can be labelled as invalid in the light of new information. A TMS can also perform default reasoning, because it can make deductions such as "Unless there is evidence to the contrary, infer A" [DEK86].

A number of points should be made regarding the gap between the vision outlined here and the current state of AI technology. A criteria for judging the suitability of a
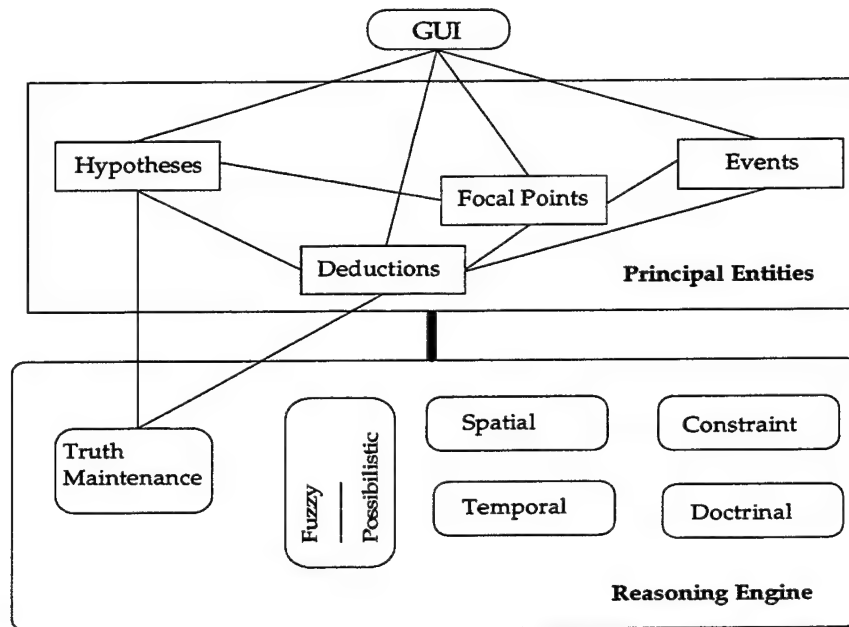

Figure 1:   Possible Components of a Mature HEDM System

representation for an application is its *expressiveness*. The more expressive a representation, the more likely it will be able to capture all the concepts necessary for unrestricted inferencing. The price, however, is generally a commensurate decrease in the representation's *tractability*, that is, its *efficiency* and *implementability*. A further difficulty may arise because the different reasoning strategies outlined probably imply a need for multiple representations. The integration of multiple representations within a single application may not be straightforward.

There are many other areas within AI where research would be likely to prove useful. These include distributed AI, blackboard architectures and architectures for complex knowledge based systems. They could also encompass natural language processing and the generation of plausible explanations. To ensure that the work is scientifically well grounded, the decision support and meta-cognition literature should also be examined.

5. The Immediate Way Ahead

For HEDM to be successfully developed, the work needs to proceed along several paths. There is a need to create a storyboard for the system, to further develop the concepts to the point where they can be properly evaluated and presented to the user community for feedback. The storyboard might then form the basis for the development of a simple concept demonstrator. It will be necessary to develop one or more simple scenarios on which to base the storyboard.

There is also a need to perform research into some of the key enabling AI technologies for this work. Three areas whose importance mark them as high priority candidates for research are truth maintenance, techniques for approximate and evidential reasoning and knowledge and data representation.

6. Conclusions

The Competing Hypotheses technique seems worthy of attention for two reasons. It is a central component of the analysis process. It also offers the potential of integrating some of the diverse tools, that are commercially available or being researched, within a single framework. The proposed work therefore has the potential to deliver major benefits to the Intelligence community within the ADF. The work can also act as a framework project in which many of the key research issues of information and data fusion can be addressed.

7. Acknowledgments

Many people within ITD have helped me develop this proposal. In particular, I thank Peter Calder, Janet Aisbett, Jeff Schapel, Mark Nelson, David Parker, Jennie Clothier and John O'Neill.

References

CBT81   "Combat Intelligence", Australian Army, Manual of Land Warfare, Intelligence Training, Vol. 1 Pamphlet No. 3, 1981.

DEK86   De Kleer, J. "An Assumption-based TMS" Artificial Intelligence 28 (1986) pp 127 - 162.

GOR94   Gori, R. and Parker, D. J., "A Data Model for the Augmented Conclusion Model", ITD Divisional Paper ITD-94-14, June 1994.

HAY83   Hayes-Roth, F., Waterman, D. A. and Lenat. D. B. (eds), "Building Expert Systems", Addison-Wesley, 1983.

HEU   Heuer, R. J., Jr. "Guidelines for Analysis of Competing Hypotheses", Notes from a lecture delivered to the U. S. Defence Intelligence College during the 1980s. Article obtained from the Defence Intelligence Wing, School of Military Intelligence, Canungra.

PRI92   Price, R., and Calder. P.,"The Expectation Model and Template Matching for Intelligence Processing", Information Technology Division Divisional Paper ITD-92-22, November 1992.

STR87   Strat, T. M. and Smith G. B., "The Core Knowledge System", Technical Report No. 426, SRI International, October 1987.

VAU90   Vaux, J., "Replicating the Expert", New Scientist, March 3, 1990, pp39 - 42.

WOO88   Woods, D. D., "Commentary: Cognitive Engineering in Complex Dynamic Worlds" in Hollnagel, E., Mancini, G., and Woods, D. D., "Cognitive Engineering in Complex Dynamic Worlds", Academic Press, 1988.

A Decision Aid for Information Integration in Naval Mine Threat Evaluation

Todd Mansell

Defence Science & Technology Organisation

Maritime Operations Division

Melbourne

Abstract

This paper describes the mine threat evaluation problem and the decision aid software being developed to assist the execution of this task. Mine threat evaluation (MTE) is the process of estimating the threat posed by sea mines to shipping in a given area. The MTE task involves examining all available evidence in order to estimate the type, and hence threat posed, by the laid minefield. Due to the large amount of disparate information available for MTE, a decision aid, Horizon, is being developed. Horizon is a generic information fusion software package for representing and fusing imprecise information about the state of the world, expressed across suitable frames of reference. The Horizon software is currently at prototype stage.

1. Introduction

The sea mine is increasingly becoming a weapon of choice for all types of countries. Mines provide a powerful political and military option at relatively low financial cost. For this reason, mine warfare is not only employed by the wealthy countries, but is also affordable to third world economies. Mines are as effective against the human mind as they are against maritime vessels. The stress created by uncertainty over mining activities unsettles the ships' command and crew, as well as tying up valuable navy resources to counter such threats.

In evaluating the threat posed by a possible minefield, one must use the available information to determine how many and what type of mines may have been laid in a particular geographic region. The information used to make these assessments is typically uncertain and/or incomplete (see section 2.5) and is likely to require a measure of confidence in its reliability. All this information should be used when estimating the type of minefield laid.

To assist the mine threat evaluation task a concept demonstration decision aid (Horizon) based on evidential reasoning (Lowrance *et al* 1991) has been developed. Evidential reasoning provides a methodology for representing and reasoning with information from disparate sources, expressed across a number of frames of reference. Horizon propagates the initial information to produce a measure of confidence in the type of minefield laid. The description of the minefield includes reference to the types of mines laid, the number of mines laid, the platform used for delivery, and the country responsible.

This paper describes the mine threat evaluation task and the evidential reasoning software designed to help carry out that assessment.

1.1 Mine Threat Evaluation

A number of steps can be taken to minimise the threat posed by mining. The first, and most obvious, is to analyse the information available indicating that mining has taken place and the threat it poses to maritime forces. This analysis constitutes the mine threat evaluation (MTE) problem and is summarised by the following questions. Has a minefield been laid? Which country is responsible? Where is the minefield located? What type of mines were laid? What platforms were used to deliver the mines?

The answers to these questions will determine the type of mine countermeasures (MCM) deployed. The specific reasoning behind the current application of MCM forces is dependent on a number of factors, one of which is the commander's perception of the threat posed by the minefield.

The information used to answer these questions is typically uncertain, incomplete, and occasionally incorrect. This requires an officer at the Local Area Commander level to evaluate all available information and best estimate the threat it poses.

2. Mine Warfare

Mine warfare dates back to 600 BC (Hartmann 1991), and has been recognised as a serious threat to maritime transportation ever since. To understand the task of mine threat evaluation, one requires some knowledge of mine warfare. This section outlines some important issues in mine warfare, including mine types, methods for deploying mines, minefield planning, and mine countermeasures. An understanding of mine warfare gives insight into the type of information used by the Local Area Commander and how it shapes his or her reasoning.

2.1 Mine Types

Mines are generally considered to be weapons that act independently[1], activated by sensing the presence of a vessel. They are often be categorised either by the way they wait for a target, or the way they are actuated (a good description of mine types is contained in Han-Chung, 1991). Ground mines are negatively buoyant and remain on the sea-bed. Moored mines are positively buoyant and are held in position by a mooring attached to a sinker. Other positively buoyant mines include drifting mines (free to move under the influence of the wind and current), creeping mines[2] (having their drifting impeded by a snag line), oscillating mines[2] (oscillating between set depths), and the rising mine (released from its sinker when activated by a vessel's influence). Then there are the mines that contain their own propulsion equipment, such

---

[1] Controlled mines do exist, but were not considered sufficiently common to be discussed.

[2] This type of mine is unlikely to be used as it contravenes the Geneva Convention on warfare.

as the mobile mine (swims like a torpedo and sinks to become a ground mine) and the homing mine (lies below the surface and homes on a target when activated).

An even more important consideration for the Local Area Commander is how the mine is actuated. This will usually be by physical contact with the mine (contact mines) or by sensing changes to the environment around the mine caused by the target (influence mines). These changes may be produced by a disturbance in the earth's magnetic field due to the presence of a metal hulled ship (ie a magnetic signature), the acoustic noise made by the ship (including machinery and flow noise), the static pressure variation caused by the displacement of the target. Alternatively, mines can be actuated by a combination of these effects, allowing better target discrimination. For example, a mine may target an ore carrier by being set to actuate in the presence of both a large static pressure and a large magnetic signature.

An understanding of how the mine is actuated is a powerful piece of information as it allows the Local Area Commander to accurately assess the threat posed by the mine. In addition this allows him or her to configure the MCM force to best reduce that threat. For example, the Local Area Commander may be expecting a bulk ore carrier in port, and may believe that magnetic ground mines could have been laid to prevent such traffic. To reduce the threat posed by these mines, he or she may tailor the MCM to target mines set to actuate on a large magnetic signature (from a large steel ship).

2.2 Mine Delivery

A valuable piece of information in mine threat evaluation is knowing the enemy's ability to deliver specific mines to your location. The country carrying out mining must have the appropriate platforms to deliver specific mine types, and those platforms must have the range (and necessary support) to deliver the mines to the relevant location. The method of delivery can be broken down into the following categories:

- Surface Delivery: Surface ships are most easily detected and identified (particularly in coastal regions), but pose a problem in that their payloads (both mine type and number laid) are difficult to infer and their operating range is often broad. However, inferences about the number of mines laid can be drawn from the type of ship, its manoeuvring, and an estimated loiter time. Further, surface vessels can place mines with a reasonable degree of accuracy, but are usually restricted to laying along the ship's track[3], giving information on the possible minefield location.

- Submarine Delivery: Submarines are able to place mines accurately and in a stealthy manner, but are restricted in the locations they can operate (particularly if harbours are protected by submarine nets). They are also limited in the number and type of mines they carry, hence inferences can be made on the number and type of mine laid. The submarine is a particularly difficult vessel to detect, and generally has an extended operating range, often resulting in the information about its presence or

---

[3] Unless mobile mines are being deployed.

activity being speculative (although submarines are not currently common in our region).

- Aircraft Delivery: Aircraft are fast and effective platforms for laying mines, but provide good information about the country carrying out mining, and the maximum number of mines laid (as aircraft have known maximum payloads). In addition, tracking information and documented aircraft operating ranges (excluding air-to-air refuelling) will be used to estimate minefield location.

Information about platform activity in an area is necessary to carry out meaningful mine threat evaluation. This information, when combined with intelligence about platforms operated by countries and their mines stockpiled, allows assertions to be made about the number and type of mines that may have been laid.

2.3 Minefield Planning

When carrying out mine threat evaluation the Local Area Commander should consider the factors that would influence the enemy's minefield planning. This will include; (1) the likely objective to be achieved by mining (eg, destruction of specific vessel types, disruption to shipping, etc.); (2) the danger posed to mine laying forces by local area defences, and detection by other forces in the region; (3) are specific geographic features of the area conducive to mining, (eg, choke points, transit lanes, ports etc.); (4) selection of a mine suited to a region, and targeting it to specific vessel types; (5) the problems involved with clearing the minefield; (5) placement of the minefield in an area likely to be traversed by the target vessels. Understanding the issues facing the mine laying forces should influence the resulting MCM operations.

2.4 Mine Countermeasures

The role of MCM is to permit warships and merchant vessels to enter and leave ports or to keep to the seas without unacceptable loss or damage due to enemy mines. This aim can be achieved by preventing the enemy from effectively laying minefields, by removing mines from required areas, by avoiding minefields, or by changing the character of ships so that they are unlikely to actuate mines.

When it is necessary to traverse waters suspected to be mined, then physically removing, exploding, or disarming mines to clear a channel for friendly shipping is required. This requires sweeping and/or hunting operations to take place. Sweeping involves towing a device that simulates the influence field of a ship[4] thereby causing the mine to explode, for example, a noise maker to actuate acoustic mines, and/or dyads to trigger magnetic mines, (sweeping moored mines involves towing a device that cuts the mooring, bringing the mine to the surface for detonation). However, sweeping is a slow and expensive exercise that does not guarantee all mines have been neutralised[5].

---

[4] This influence field can be optimised to appear as the expected target vessel, or the vessel requiring safe passage.

[5] Mine may be programmed with a ship-count that tells the mine to detonate when the nth target vessel passes in range (ie. on the nth actuation).

Rather, it reduces the probability of ships actuating mines to a level that is acceptable to the command responsible for ordering forces into mined waters.

Mine hunting is the active searching for mines, and their subsequent disposal. This usually involves using sonar to detect mine-like objects, at which time a mine disposal vehicle or clearance diver is sent down to classify and neutralise the mine. Limitations of the sensors (due to physical constraints of the environment) make hunting a slow process, although given good bottom conditions it may not necessarily take any longer than sweeping, and is not sensitive to ship counts.

Clearance diving and passive measures are other commonly employed MCM strategies. Clearance divers are specially trained in methods for locating and removing mines in shallow water, while passive measures involve using things like degaussing coils to reduce a ship's disturbance in the earth's magnetic field, or other tactics to reduce the acoustic and pressure disturbances produced by vessels.

2.5 Information Sources

The information used by a Local Area Commander for mine threat evaluation is often uncertain, incomplete and occasionally conflicting. Table 1 summarises the type of information likely to be supplied to the local area commander. In this table, *hard intelligence* refers to the type of information collected over a long period of time and stored in a defence intelligence organisation's database; *enemy doctrine* describes what is known about how the enemy operates: *Electronic surveillance* is intelligence collected by sensors; *Human surveillance* is reports provided by people in the field; *Maritime intelligence* refers to the information processed by a shore based maritime intelligence centre.

As an example, when estimating the number of mines in the minefield, one might make inferences based on the type of platforms operating in the area. Reports may state an unknown enemy submarine may be operating in the area, and there may have been several reports of between 2 and 4 fighter-bomber aircraft active around the harbour entrance over the last 48 hours. Hard intelligence tells us that Red country is likely to carry out mining and have the ability to operate the observed platforms in our area. We also know that Red country is able to deliver two Mk 48 or MP-80 mines from each fighter-bomber and up to six Mk 48s from their submarine. Using this information one may determine that the evidence points to a minefield of between 4 and 14 magnetic or acoustic/magnetic ground mines located at the harbour entrance transit lane by Red country. The weight one places on each piece of evidence will influence how the various combinations of mine number and type are ranked.

3. Evidential Reasoning

Evidential reasoning (Lowrance *et al* 1991) is a formalism for representing information from disparate sources that is expressed in different frames of reference, and provides techniques to manipulate that information. Evidential reasoning (E-R) is an extension of Shafer's (1976) work on belief functions (called Dempster-Shafer Theory). Being a departure from classical probability theory, E-R uses information that is typically uncertain, incomplete and error-prone. E-R maintains the association between the

Table 1: Information sources and the type of information they provide.

Information Type	Comments
Hard Intelligence	• Country likely to carry out mining based on the political relations. • Mine laying capabilities of foreign forces, (eg, readiness, capabilities, and geographic limitations). • Platforms available for mine warfare (number and type). • Platform payload figures (number and type of mines they can deliver, as well as mine laying rates). • Estimated number and type of mines stock piled. • Reports on planned mining operations.
Enemy Doctrine	• Geographic features likely to be exploited by mining. • Strategies and tactics employed in mine warfare.
Electronic Surveillance	• Air traffic control. • JORN (Jindalee Over the Horizon Radar). • Communications intelligence. • Detection of emissions from enemy platforms. • Satellite information provided by the ADF or our allies.
Human Surveillance	• Mine watch organisations. • Coast watch organisations. • Strategically positioned spotters.
Maritime Intelligence	• Contacts logged by ADF platforms (including ESM or visual). • Reconnaissance operations. • Contacts reported by other friendly platforms. • Spurious reports of enemy activity.

measure of belief and disjunctions of events rather than forcing probabilities to be distributed across atomic possibilities. The result is that one need no longer assume that all data are available and being utilised.

E-R is used to assess the effect of all pieces of available evidence on a hypothesis, making use of domain-specific knowledge. A propositional space called the *frame of discernment* (or frame) is used to define a set of basic statements, exactly one of which may be true at any one time, and a subset of these statements is defined as a *propositional statement*. For example, a frame, θ_A, may be used to represent the platform used to deliver mines (this implementation of the MTE domain currently uses six frames to describe the environment as shown in Figure 1).

Once frames have been established, bodies of evidence (BOE) are use to make probabilistic assessments about the confidence in propositional statements relative to the frame. Belief assigned to non-atomic propositional statements explicitly represents the lack of information available to resolve between the propositions, resulting in a distribution appropriate to the granularity of the evidence. For example, if the evidence does not distinguish between fighter or fighter-bomber aircraft, then the evidence is

attributed to the disjunction of the propositions (probability theory would require the evidence be divided between individual propositions).

E-R provides a complete methodology for information integration, including the collection of information in its native frame of reference, discounting (due to the credibility of the source), translation to a related frame, projection into the future (or past), and fusion with other independent BOEs.

Compatibility relations are used to characterise interrelationships between different propositional spaces. This allows reasoning to be carried out on information described at different levels of abstraction or on frames of reference with overlapping attributes. Figure 1 shows all the frames used in the MTE problem, with a link between two frames representing the existence of a compatibility relation. In this domain, a compatibility relation between the *platforms* and *mine types* frames, describes what is known about the types of mines that can be deployed by certain platforms. Therefore, evidence about the type of platform provides information about the mine type laid, and vice versa.

E-R uses Dempster's rule of combination (Lowrance *et al* 1991) to fuse multiple independent BOEs into a single BOE, emphasising points of agreement and deemphasising points of disagreement. Dempster's rule is both commutative and associative (evidence can be combined in any order) providing a consensus of what was disparate opinion.

The selection of this method for dealing with uncertainty was not based on competency, as probability theory and fuzzy logic are very capable of representing uncertain information. Instead, E-R was eventually selected over probability theory for its natural representation and manipulation with information contained at different levels of abstraction and in different frames of reference. Probability theory (like statistical inference and fuzzy logic) does not have an equivalent natural method for handling abstraction.

3.1 Independence of Evidence

The concept of independence is controversial in the areas of E-R and probability theory (Dawid 1979). Often centring around the areas of experimental independence or conditional independence, these theories have tended to handle dependence inadequately (Pearl 1988, Shafer 1981, Walley 1991, Kahneman, *et al* 1982). These criticisms are usually based on the difficulty of acquiring the appropriate evidence values, and applying an independence test to the data.

It has been proposed that conditional independence between BOEs is not sufficient to guarantee the validity of Dempster's fusion algorithm (Voorbraak 1991). However, a convincing argument or counter-intuitive example has not been presented to substantiate these claims. Hence, when using E-R one makes the following two assumptions about the BOEs being fused using Dempster's rule:

- the human operator can determine whether BOEs are based on the same observations, and are therefore dependent (eg, two intelligence reports quoting the same source are not independent).
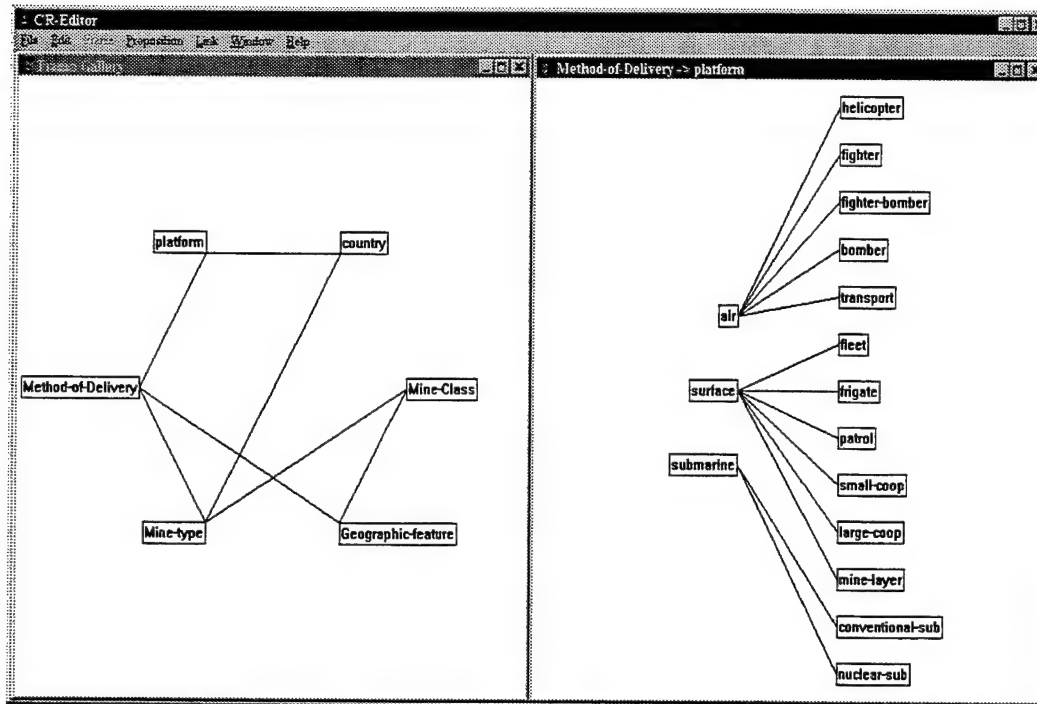
Figure 1: Example of the CR-Editor, the left window showing the frames that are linked by compatibility relations, and the right window displaying the relation between method of delivery and platform.

- people (i.e., the expert and knowledge engineer) can accurately and confidently determine whether two events or actions are independent, as proposed by Pearl (1988), Shafer (1976, and 1981), Walley (1991), and Spiegelhalter and Lauritzen (1990).

With the increasing number of successful applications of E-R to real world problems (including submarine tracking, and naval intelligence analysis (Lowrance *et al* 1991)), and a lack of negative experiences, E-R is considered suitable to the MTE information fusion task.

4. Horizon Program

The software package called Horizon[6] (currently at the prototype stage) is a domain-independent E-R system that is currently being applied to the MTE domain. One challenge in developing this information-fusion software package is to make sure the design does not require the user to understand the intricacies of E-R (a goal we are still working towards). However, it is anticipated that a certain amount of understanding

---

[6] Horizon is defined in the Webster's Dictionary as the fullest range or widest limit of perception, interest, appreciation, knowledge, or experience.

(training) is required to distribute evidence in an E-R manner, as well as ensure the information is independent.

Horizon is a decision-aid program that requires a knowledge engineering process to take place before it can be applied to a problem. This involves capturing the domain by first establishing the frames of reference used to represented BOEs, and generating the compatibility relations between those frames. The amount of knowledge engineering required will depend on the domain under investigation. The MTE domain consists of 6 frames of reference used to describe the country responsible for mining, the platforms used to lay mines, the method of delivery (an abstraction of the platforms frame), mine type, mine class, and minefield's geographic location. The compatibility relations are constructed with the aid of an expert, and represent the frames of reference in which one expects information to arrive, or may require conclusions to be presented.

Horizon provides a graphical user interface for the real time display and editing of compatibility relations (called the CR-Editor, displayed in Figure 1). The CR-Editor has two types of windows. The *Frame Gallery* window links frames for which compatibility relations exist. The *CR-windows* display the compatibility relations between two frames of reference with the propositions of each frame lined up on each side. Links between these propositions represent information in one frame of reference that is simultaneously true in the other frame of reference. For example, the left window in Figure 1 displays the MTE domain frames that have compatibility relations, while the right window displays the compatibility relations that exist between the platforms and method-of-delivery frames.

Information collected from the environment (section 2.5) can be entered into the system in three ways. Firstly, static information (such as mine warfare doctrine) that does not change rapidly can be stored as BOE data files, and read into Horizon when the system is initialised. Secondly, dynamic information can be entered automatically into Horizon's database by sensors, signal processing units, expert systems, etc. Finally, other dynamic information (such as surveillance reports) can be entered directly into the system as it arrives using the window shown in Figure 2. This requires the user to select the frame of reference, then distribute belief among the listed propositions. The interface window is also used to edit and update all forms of information when required.

Horizon represents and manipulates BOEs in an object oriented manner. Each BOE is stored in its native frame of reference, where it can be selected to be included in a calculation. At present the calculation operations include *discount* (reduces the confidence in a BOE), *translate* (move to a new frame), and *fuse* (combine BOEs). Once the user has selected the BOEs to be included in the calculation, the operation is chosen. If *discount* is selected the user supplies a discount rate (a percentage between 1 and 100) at which time Horizon produces a secondary[7] BOE with a modified belief distribution. If the *translate* or *fuse* operations are selected, the user is prompted to choose the frame in which the conclusion should be expressed. The system will carry

---

[7] A secondary BOE is a body of evidence that is generated through the manipulation of initial BOE(s).
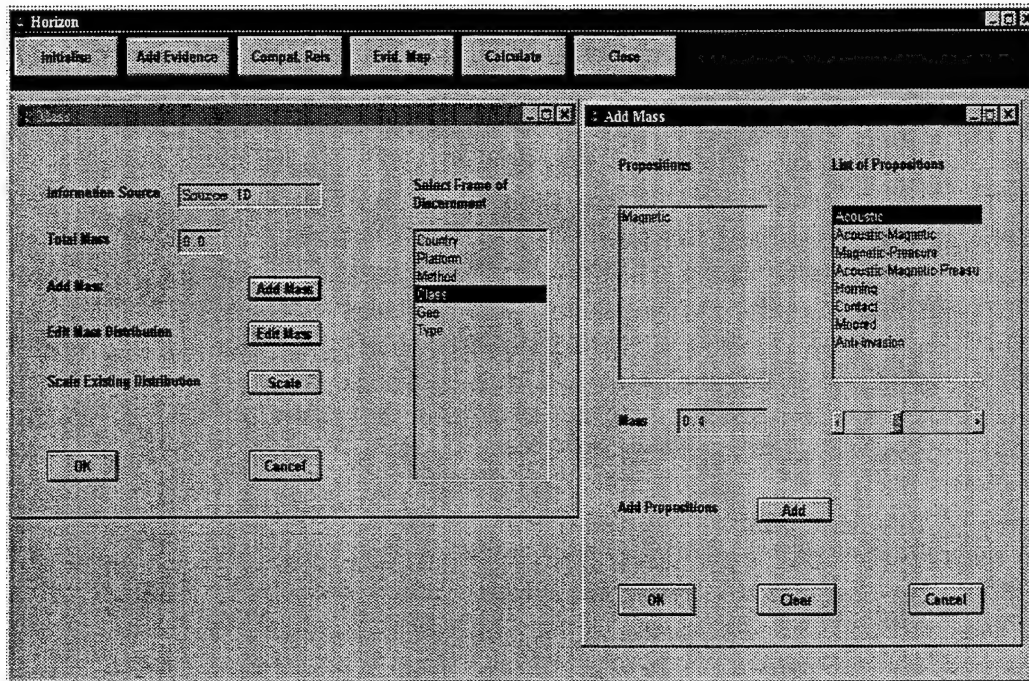
Figure 2: Windows used to create or edit a new BOE.

out the necessary operation, and in the fusion case, present the resulting BOE in the display window (Figure 3). The display window presents the pooled evidence for and against all non-zero propositional statements, as well as a measure of uncertainty (being the amount of evidence that neither supports nor contradicts that statement). The automated mapping of BOEs from their initial frames to the concluding frame is currently being written (at present the user must direct the translation of each BOE to the concluding frame).

Horizon is written in Allegro Common Lisp, with the user interface being written in PC/CLIM, making the package portable to either PC or Unix machines. The next generation of the project is due for completion by the end of 1995, and will result in an object-oriented evidential-reasoning decision aid. This version of Horizon will boast an improved user interface for the entry and display of evidence, as well as a graphical interface for the manipulation of BOEs. Future work is planned including an explanation facility based on sensitivity analysis, while automated temporal and spatial reasoning is also under investigation.

5. Discussion

Two fundamental objections to evidential reasoning exist in the literature. The question of the ability to determine independence of evidence (Voorbraak 1991) is at best inconclusive and is not substantiated by the existing applications. The problem of computational complexity is easily worked around by limiting the size of frames, usually by reducing large frames to a number of smaller frames through abstraction or using singleton frames (Gordon and Shortliffe 1985).
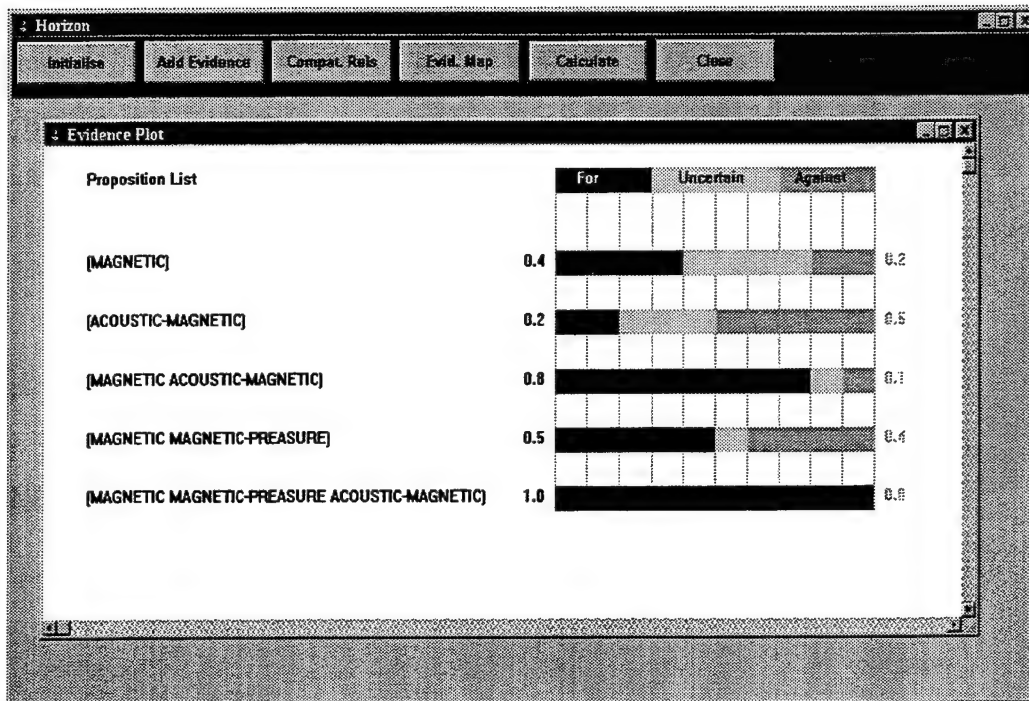
Figure 3: Window used to display the result of fused BOEs. This window presents the amount of evidence supporting, uncommitted (uncertain), and contradicting all non-zero propositional statements.

Initial results from the MTE domain indicate that independence is not a serious concern for most sources of evidence. However, this may not be the case when dealing with intelligence reports, as it is often difficult to determine independence of sources (particularly when the basis of the reports are unknown). One solution may be to require the human expert (intelligence officer) to manually fuse information suspected of being dependent before it is presented to Horizon. It is also anticipated that algorithms for fusing dependent evidence (Shi *et al* 1993) will need to be further investigated.

Horizon has been trialed on a limited set of synthetic data to demonstrate the suitability of evidential reasoning to information fusion. On this data (up to 12 BOEs) the fusion runs at near real time, with frames containing between 3 and 15 propositions, (and an average size of 8.6). The response time for fusion of large numbers of BOEs is currently under investigation.

6. Conclusion

This paper describes the mine threat evaluation problem and the Horizon decision aid currently under development at MOD. Preliminary examination of this software demonstrates that evidential reasoning is an appropriate technique for dealing with high level information fusion. Horizon is intended for use in mission planning scenarios where time can be taken to enter information into the decision aid software. However, it is not acceptable to require a naval officer to enter excessive data into a system, as

the decision aid then become a burden. For this reason, ways to reduce the amount information manually entered into the system are to be investigated.

7. Bibliography

ATP-6(B) Volume 1, Mine Warfare Principles. NATO CONFIDENTIAL.

A. P. Dawid (1979), Some Misleading Arguments Involving Conditional Independence. *Journal of The Royal Statistical Society: Series B*, 41(2): 1-31.

J. Gordon and E. H. Shortliffe (1985), A Method for Managing Evidential Reasoning in a Hierarchical Hypothesis Space. *Artificial Intelligence*, 26: 323-357.

L. Han-Chung (1991), *Using Expert Systems in Mine Warfare.* Masters Thesis, Naval Postgraduate School, Monterey, California.

G. Hartmann (1991), *Weapons That Wait: Mine Warfare in the US Navy*, Naval Institute Press, Annapolis, USA.

D. Kahneman, P. Slovic, and A. Tversky (eds) (1982), *Judgement Under Uncertainty: Heuristics and Biases*, Cambridge University Press, Cambridge.

J. Lowrance, T. Strat, L. Wesley, T. Garvey, E. Ruspini and D. Wilkins (1991), *The Theory, Implementation, and Practice of Evidential Reasoning.* Technical report, Artificial Intelligence Centre, SRI International, California.

J. Pearl (1988), *Probabilistic Reasoning In Intelligent Systems.* Morgan Kaufmann Publishers, Inc., California.

G. Shafer (1976), *Mathematical Theory of Evidence.* Princeton University Press, Princeton, New Jersey, USA.

G. Shafer (1981), Constructive Probability. *Synthese*, 48:1-60.

S. Shi, M. Hull, D. Bell, and J. Guan (1993), Combining Evidence in Evidence Theory. In *Proceedings of the First Australian and New Zealand Conference on Intelligent Information Systems*, Perth, Australia, 142-146.

D. J. Spiegelhalter and S. L. Lauritzen (1990), Sequential Updating of Conditional Probabilities On Directed Graphical Structures. *Networks*, 20(5):579-605.

F. Voorbraak (1991), On the Justification of Dempster's Rule of Combination, *Artificial Intelligence*, 48:2, 171-198.

P. Walley (1991), *Statistical Reasoning With Imprecise Probabilities.* Chapman and Hall, London.

Finding Complex Targets in Complex Scenes using Machine Learning Techniques: a viable Surveillance Paradigm?

Terry Caelli[1]   Walter F. Bischof[2]

[1]Department of Computer Science, Curtin University of Technology, GPO Box U1987, Perth, WA 6001, Australia, Email: tmc@cs.curtin.edu.au
[2]Department of Psychology, University of Alberta, Edmonton, Alberta, T6G 2E9, Canada, Email: wfb@psych.ualberta.ca

Abstract

In this paper, we discuss automatic rule generation techniques for learning relational properties of 2-D visual patterns and 3-D objects from training samples where the observed feature values are continuous. In particular, we explore a new conditional rule generation method that defines patterns (or objects) in terms of ordered lists of bounds on unary (single part) and binary (part relation) features. The technique, termed Conditional Rule Generation (CRG), was specifically developed to integrate the relational structures of graph representations of patterns and the generalization characteristics of Evidenced-based Systems (EBS). CRG takes into account the label-compatibilities that should occur between unary and binary rules in their very generation, a condition that is, generally, not guaranteed in well-known Rule Generation and Machine Learning techniques as they have been applied to problems in Computer Vision. We show how this technique applies to the recognition of complex targets and of objects in scenes, and we show the extent to which the learned rules can identify patterns and objects that have undergone non-rigid distortions.

1 Introduction

To develop systems which can detect relatively complex patterns or objects in complex scenes requires efficient and robust techniques for describing patterns and searching for them in such data structures. Machine learning, as it applies to the detection, recognition and surveillance of scenes, provides methods for solving such problems. In particular, in this paper we address the issue of just how ML is used in the following sub-system domains of:

- Feature Selection: The automatic selection and/or ordering of encoded features that can optimize the recognition processes.

- Generalization: The automatic generation of "structural descriptions" of targets that can cover a range of training pattern examples, as well as distorted and unseen examples.

- Efficiency: The optimization of search and matching procedures.

These goals can be attained, with differing degrees of success, using a wide variety of representations, learning and matching technologies.

The type of representation most frequently used in vision has been the *relational structure* (RS) where patterns are encoded as parts (graph vertices) and part relations (graph edges), both being described by a set of attributes or features. Such graph representations are limited in the sense that generalization in terms of either new views or non-rigid transformations of objects are difficult to represent. Further, pattern recognition typically involves graph matching, with a computational complexity that exponentiates with the number of parts [1, 2]. Little attention has been paid to the design of optimal search procedures that use conjoint feature states (i.e. conjunctions of particular sets of feature values) to define important characterizations of patterns, and they are less than ideal for the recognition of objects embedded in scenes. Typically, they use prior knowledge to prune the search space, as has been explored by a number of authors (for example, [3, 4]).

In contrast to the RS representation and associated constraint-based graph matching (tree search) methods, evidenced-based systems (EBS) provide a different approach to the recognition problem. Like RS, EBS works within the Supervised Learning (Learning from Example) paradigm and require subprocesses for encoding, segmentation and part/relational feature extraction. Patterns and objects are encoded by rules of the form:

if {**condition**} then {**evidence weights for each class**}

where the rule condition is usually defined in terms of bounds on feature values, and where rules instantiated by data activate weighted evidence for different pattern classes. Such rules can be defined over pattern features of arbitrary arities and the main problem in EBS has been to determine the feature bounds and evidence weights. That is, EBS typically involves partitioning feature spaces into regions associated with different pattern classes, and the problem has been to determine classification rules that attempt to minimize misclassification while, at the same time, maximizing rule generalization. Since these regions are not necessarily class disjoint, evidence weights are typically used to index the degrees to which samples within the region correspond to different classes. "Generalization" is then defined by the associated volumes of the regions that define the rules in feature space.

Evidence weights are typically derived from the relative frequencies of different classes per region [5] or, more recently, by minimum entropy and associative neural network techniques [6]. In either case, the label-compatibilities of data parts and their relations were only encoded through the simultaneous activation of both unary and binary rules.

Although such systems allow generalizations from samples, they only attain implicit learning of the RS, in so far as unary rules (rules related to part features) and binary rules (rules related to part relational features) are both activated to evidence patterns or objects. EBS do not explicitly consider the compatibility between unary and binary rules as they reference specific pattern parts and their relations. Indeed, patterns are uniquely defined by the enumeration of specific *labeled* unary and binary feature states of the form $U_i - B_{ij} - U_j$. The two patterns shown in Figure 1 have isomorphic unary and binary feature states
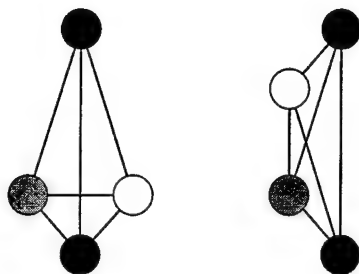
Figure 1: Two patterns with isomorphic unary (U = vertex color and orientation) and binary (B = line length and orientation) feature states but differing in their label-compatibilities: the sequences of $U_i - B_{ij} - U_j$... differ between the two patterns).

but are not identical. This shows that the existence of such correspondences does not guarantee identity in shape unless the unary and binary feature labels are compatible. Even given this, determining the uniqueness of a pattern may involve checking for attribute and label consistencies of higher order than, say, the consistencies of isolated parts or part-relation pairs. Rules satisfying the "label compatibility" property of rules must evidence specific objects or patterns uniquely, i.e. lists of unary and binary feature states must evidence *specific joint occurrences of parts and relations*. The problem then is how rules having this property can be generated automatically.

As already stated, the simplest representation for visual patterns that takes into account the label-compatibility of unary and binary features, is a graph. Graph matching techniques are used to solve the recognition problem where a sample pattern structure (for example, new data for classification) is matched to a model structure by searching for a label assignment that maximizes some objective similarity function [2]. Pattern classes are represented by sets of instances and classification is thus achieved by searching through all model graphs to determine the one producing the best match. This representation and graph matching approach, in the form of interpretation trees and feature indexing, has been the preferred architecture for object recognition [4, 7].

Different approaches to improving the efficiency of the matching processes have been proposed, such as constraint-based decision trees [3], "pre-compiled" tree generation [8], heuristic search techniques [9], dynamic programming [10], relaxation labeling [11] or hierarchical model fitting [12]. However, the problem of learning and constructing union and discrimination trees for structural descriptions has been addressed only sporadically in the literature, such as in [13] within the framework of inductive learning of symbolic structural descriptions or in [14] within the framework of probabilistic inductive prediction of sequential patterns.

In summary, graph matching methods solve the label-compatibility problem but do not allow for efficient representation of pattern classes via union and discrimination trees. Further, such representations and algorithms do not consider a fundamental issue in pattern recognition, *generalization*, i.e. the ability for the system to recognize equivalences between patterns that are not identical. Also, they do not fully exploit learning to determine the optimal search path amongst

unary and binary feature states to evaluate the existence of specific patterns. For example, in 3D object recognition, it is often necessary to classify objects as belonging to a specific object type even though individual samples of the class may be non-rigid transformations of other members of the same class - as in different types of coffee mugs, etc. At the same time, we wish to automatically generate descriptions of 3D objects that not only enable such generalizations but also do so with respect to the description length of the rules (the length of strings of unary-binary-unary-... feature bounds). Evidence-based systems provide for generalization, but do not adequately address the label-compatibility problem.

In the following Sections we focus on the analysis of a new technique for the learning of structural relations, **Conditional Rule Generation** (CRG). It generates a tree of hierarchically organized rules for classifying structural pattern descriptions that aims at "best" generalizations of the rule bounds with respect to rule length (the number of U-B-U, etc., conditional feature lists). The aim of this paper is to show how the technique can be used to solve problems involving the recognition of 2D patterns and 3D objects in complex visual scenes.

2　The Conditional Rule Generation Method

In CRG, rules are defined as clusters in Conditional Feature Spaces which correspond to either unary or binary features of the training data. The clusters are generated to satisfy two conditions: one, they should maximize the covering of samples from one class and, two, they should minimize the inclusion of samples from other classes. In our approach, such rules are generated through controlled decision tree expansion and cluster refinement as described below.

2.1　Cluster Tree Generation

Each pattern (a 2D sample pattern or a view of a 3D object) is composed of a number of parts (pattern components) where, in turn, each part $p_r, r = 1, ..., N$ is described by a set of unary features $\vec{u}(p_r)$, and pairs of parts (p_r, p_s) belonging to the same sample (but not necessarily all possible pairs) are described by a set of binary features $\vec{b}(p_r, p_s)$. Below, $S(p_r)$ denotes the sample (in 3D object recognition, a "view") a part p_r belongs to, $C(p_r)$ denotes the class (3D object recognition - object) $S(p_r)$ belongs to, and H_i refers to the information, or cluster entropy statistic:

$$H_i = -\sum_k q_{ik} \ln q_{ik} \qquad (1)$$

where q_{ik} defines the probability of elements of cluster i belonging to class k. We first construct the initial unary feature space for all parts over all samples and classes $U = \{\vec{u}(p_r), r = 1, .., N\}$ and partition this feature space into clusters U_i. In our approach, the initial clustering procedure is not critical, as will be discussed further below. Clusters that are unique with respect to class membership (with entropy $H_i = 0$) provide a simple classification rule for some patterns (e.g. U_3 in Figure 2). However, each non-unique (unresolved) cluster U_i is further analyzed with respect to binary features by constructing the (conditional) binary feature space $UB_i = \{\vec{b}(p_r, p_s) \mid \vec{u}(p_r) \in U_i$ and $S(p_r) = S(p_s)\}$. This feature space is clustered with respect to binary features into clusters UB_{ij}. Again, clusters that are unique with respect to class membership provide classification
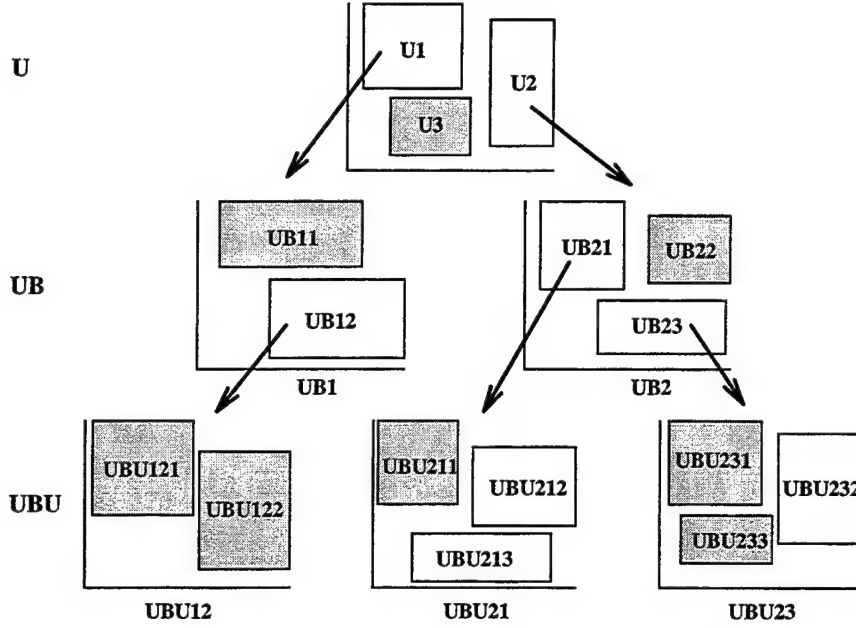
Figure 2: Cluster Tree generated by the Conditional Rule Generation Procedure (CRG). The unresolved unary clusters (U_1 and U_2) - with element from more than one class - are expanded to the binary feature spaces UB_1 and UB_2, from where clustering and expansion continues until either all rules are resolved or the predetermined maximum rule length is reached, in which case rule splitting occurs.

rules for some objects (e.g. UB_{11} in Figure 2). Each non-unique cluster UB_{ij} is then analyzed with respect to unary features of the second part and the resulting feature space $UBU_{ij} = \{\vec{u}(p_s) \mid \vec{b}(p_r, p_s) \in UB_{ij}\}$ is clustered into clusters UBU_{ijk}. Again, unique clusters provide class classification rules for some objects (e.g. UBU_{121} in Figure 2), the other clusters have to be further analyzed, either by repeated conditional clustering involving additional parts at levels $UBUB$, $UBUBU$, etc. or through cluster refinement, as described below.

Each element of a cluster at some point in the cluster tree corresponds to a sequence $U_i - B_{ij} - U_j - B_{jk}...$ of unary and binary features associated with a non-cyclic sequence (path) of pattern parts. In the current implementation, we analyze all path permutations in order to guarantee classification of arbitrary partial patterns, even though this leads to the generation of redundant set of rules. Elsewhere, we have studied ways of reducing this redundancy through the use of feature ordering [15].

In the current implementation of CRG, we have used a simple splitting-based clustering method to enable the generation of *disjoint* rules and to simplify the clustering procedure. Cluster trees are generated in a depth-first manner up to a maximum level of expansion. Clusters that remain unresolved at that level are split in a way described in the following Section.

2.2 Cluster Refinement

All non-unique (unresolved) clusters remaining at a given level of the cluster-tree generation (e.g. clusters UBU_{212}, UBU_{213} and UBU_{232} in Figure 2) have to be analyzed further to construct unique decision rules. One way of doing this is to simply expand the cluster tree, analyzing unary and binary attributes of additional parts to generate rules of the $\{UBUB...\}$ form. However, this may never give completely "resolved" branches in the cluster tree. Alternatively, the derived clusters in the tree can be refined or broken into smaller clusters, using more discriminating feature bounds, as described below. Both approaches have their respective disadvantages. Cluster refinement leads to an increasingly complex feature-space partitioning and thus may reduce the generality of classification rules. Cluster-tree expansion, on the other hand, successively reduces the possibility of classifying pattern fragments, or, in 3D object recognition, classifying objects from partial views. In the end, a compromise has to be established between both approaches.

In cluster refinement, two issues must be addressed, the refinement me-thod and the level at which cluster refinement should be performed. Consider the cluster tree shown in Figure 2 with non-unique clusters UBU_{212}, UBU_{213} and UBU_{232}. One way to refine clusters (for example, cluster UBU_{232}) is to re-cluster the associated feature space (UBU_{23}) into a larger number of clusters. However, classification rules associated with other clusters (UBU_{231} and UBU_{233}) are lost and have to be recomputed. Alternatively, given that each cluster is bounded by a hyper-rectangle in feature space, refinement of a cluster can be achieved by splitting this rectangle along some optimal boundary. This ensures that other sibling clusters remain unaffected. With respect to the level at which cluster refinement is performed, instead of splitting an unresolved leaf cluster (UBU_{232}) one could split any cluster in the chain of parent clusters (UB_{23} or U_2).

Consider splitting the elements of an unresolved cluster C along a (unary or binary) feature dimension F. The elements of C are first sorted by their feature value $f(c)$, and then all possible cut points T midway between successive feature values in the sorted sequence are evaluated. For each cut point T, the elements of C are partitioned into two sets, $P_1 = \{c \mid f(c) \leq T\}$ with n_1 elements and $P_2 = \{c \mid f(c) > T\}$ with n_2 elements. We define the partition entropy $H_P(T)$ as

$$H_P(T) = n_1 H(P_1) + n_2 H(P_2). \tag{2}$$

the cut point T_F that minimizes $H_P(T_F)$ is considered the best point for splitting cluster C along feature dimension F (see also [16]). The best split of cluster C is considered the one along the feature dimension F that minimizes T_F. As noted above, rather than splitting an unresolved leaf cluster C_L, one can split any cluster C_i in the parent chain of C_L. For each cluster C_i, the optimal split T_F is computed, and the cluster C_i that minimizes T_F is considered the optimal level for refining the cluster tree. Clusters above C_L may contain elements of classes other than those that are unresolved in C_L. Hence, in computing H_P for those clusters, we consider only elements of classes that are unresolved in C_L.

Two further properties of the splitting procedure are important, since they affect the type of rules generated by CRG. First, if a nonterminal cluster of the cluster tree is split, the feature spaces conditional upon that cluster are recomputed since the elements of the feature space have changed. Second, in the case of a tie, i.e. if two or more clusters have the same minimal partition

entropy $H_P(T)$, the cluster higher in the cluster tree is split. Together, this leads to CRG having a clear preference for shallow cluster trees and for short rules, which, in turn, leads to efficient rule evaluation.

The rules generated by CRG are sufficient for classifying new pattern or pattern fragments, provided that they are sufficiently similar to patterns presented during training and provided that the patterns contain enough parts to instantiate rules. However, cluster trees and associated classification rules can also be used for partial rule instantiation. A rule of length m (for example, a *UBUBU*-rule) is said to be partially instantiated by any shorter $(l < m)$ sequence of unary and binary features (for example, a *UBU*-sequence). From the cluster tree shown in Figure 2, it is clear that a partial instantiation of rules (for example, to the *UB*-level) can lead to unique classification of certain pattern fragments (for example, those matched by the U_3 or UB_{11} rules, but it may also *reduce* classification uncertainty associated with other nodes in the cluster tree (for example, UB_{23}). From the empirical class frequencies of all training patterns associated with a node of the cluster tree (for example, UB_{23}), one can derive an expected classification vector, or *evidence* vector. The evidence vector is used to predict the classification vector of any part, or sequence of parts, that instantiates the associated rule.

In summary, CRG has been specifically developed to enable the learning of patterns defined by parts and their relations. The technique determines the type of inductive learning (attribute generalizations) that can be performed and the associated minimum length descriptors of shapes for recognition. Finally, since the method precompiles patterns as relational trees, the technique is ideally suited for the learning of patterns with variable complexity and their detection in scenes.

3   Detecting 2D Patterns in Scenes

In this Section, we illustrate learning of 2D patterns using the CRG method, the recognition of these patterns embedded in more complex scenes using the rules generated by CRG. The example, **line triples**, consists of four classes of patterns with four training examples each (see Figure 3a). Each pattern is described by the unary features "length" and "orientation", and the binary features "distance of line centers" and "intersection angle". The line patterns are simplified versions of patterns found in geomagnetic data that are used to infer the presence of certain metals or minerals.

CRG was run with maximum rule length set to $maxlevel = 5$ (i.e. rules up to the form of *UBUBU* are being generated), and it produced 35 rules, 3 *U*-rules, 18 *UB*-rules, 2 *UBU*-rules, and 12 *UBUB*-rules.

At recognition time, a montage of patterns was presented (see Figure 3b), and the patterns were identified and classified as described below, producing the classification result shown in Figure 3d. Pattern identification and classification was achieved using the following steps:

1) Unary features are extracted for all scene parts (lines), and binary features are extracted for all adjacent scene parts, i.e. pairs whose center distance does not exceed a given limit. The adjacency graph is shown in Figure 3c, where dots indicate the position of the line centers, and adjacent pattern parts (lines) are connected.
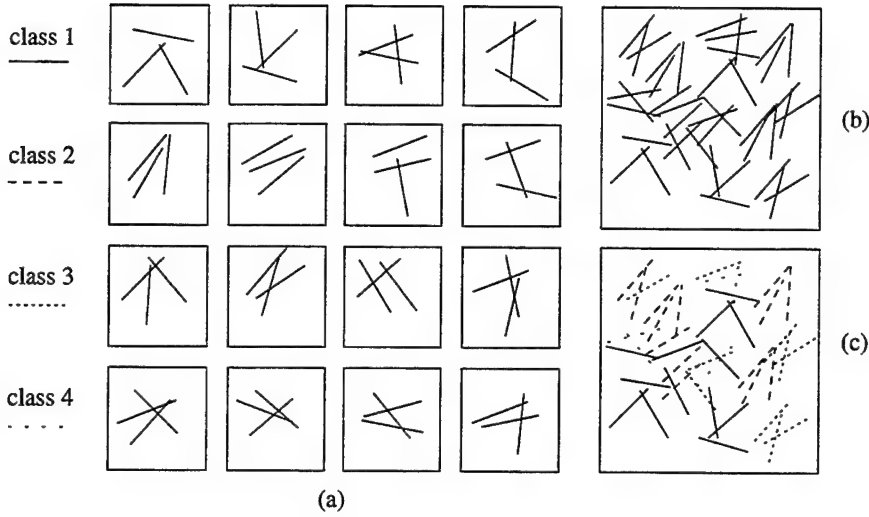
Figure 3: (a) Four classes of patterns with four training patterns (views) each. Each pattern is composed of three lines. Lines are described by the unary features "line length" and "orientation", and pairs of lines are described by the binary features "distance of line centers" and "intersection angle". (b) Montage of (slightly distorted) line triples. (c) In the adjacency graph for the montage, dots indicate the position of the line center and adjacent lines (with a center distance below a given limit) are connected. (d) Result of the pattern classification using the rules generated by CRG. Class labels for each line are shown on the right.

2) Given the adjacency graph, all non-cyclic paths up to a certain length l are extracted, where $l \leq maxlevel$. These paths, termed *chains*, constitute the basic units for pattern classification. A chain is denoted by $S = < p_i, p_j, ..., p_n >$ where each p_i denotes a pattern part. For some chains, all parts belong to a single learned pattern, but other chains are likely to cross the "boundary" between different patterns.

3) Each chain $S = < p_i, p_j, ..., p_n >$ is now classified using the classification rules produced by CRG. Depending on the unary and binary feature states, a chain may or may not instantiate one (or more) classification rules. In the former case, rule instantiation may be partial (with a non-unique evidence vector $\vec{E}(S)$), or complete (with $H[\vec{E}(S)] = 0$). As discussed above, the evidence vector for each rule instantiation is derived from the empirical class frequencies of the training examples.

4) The evidence vectors of all chains $< p_{i_1}, p_{j_1}, ..., p_n >$, $< p_{i_2}, p_{j_2}, ..., p_n >$, etc., terminating in p_n determine the classification of part p_n. Some of these evidence vectors may be mutually incompatible and others may be non-unique (through partial rule instantiation). Here, we have studied two ways of combining the evidence vectors, a winner-take-all solution and a relaxation labeling solution.

Implementation of the **winner-take-all** (WTA) solution is straightforward. The evidence vectors of all chains terminating in p_n are averaged to give $\vec{E}_{av}(p_n)$,

and the most likely class label is enacted. However, the WTA solution does not take into account that, for a chain $S = <p_i, p_j, ..., p_n>$, the average evidence vectors $\vec{E}_{av}(p_i)$, $\vec{E}_{av}(p_j)$, ...,$\vec{E}_{av}(p_n)$ may be very different and possibly incompatible. If they are very different, it is plausible to assume that the chain S is "crossing" boundaries between different patterns/objects. In this case, the chain and its evidence vectors should be disregarded for the identification and classification of scene parts.

This is achieved in the **relaxation labeling** (RL) solution, where evidence vectors are weighted according to intra-chain compatibility. Specifically, the RL solution is given by

$$\vec{E}^{t+1}(p_i) = \Phi \left[\sum_{S=<p_i...p_n>} \vec{E}^t(p_i) C(p_i, p_n) \right] \tag{3}$$

where $\vec{E}^t(p_i)$ corresponds to the evidence vector of p_i at iteration t, with $\vec{E}^0(p_i) = \vec{E}_{av}(p_i)$. $C(p_i, p_n)$ corresponds to the compatibility between parts p_i and p_n, and Φ is the logistic function

$$\Phi(z) = (1 + \exp[-20(z - 0.5)])^{-1}. \tag{4}$$

Further, we have encoded the compatibility function in terms of the scalar product between the evidence vectors of parts p_i and p_n,

$$C(p_i, p_n) = \vec{E}(p_i) \cdot \vec{E}(p_n). \tag{5}$$

For identical evidence vectors $\vec{E}(p_i)$ and $\vec{E}(p_n)$, $C(p_i, p_n) = 1$, and for incompatible evidence vectors, for example $\vec{E}(p_i) = [1, 0, 0]$ and $\vec{E}(p_n) = [0, 1, 0]$, $C(p_i, p_n) = 0$.

Compatibility of evidence vectors is a weak constraint for updating the evidence vectors of each part and it may even have an adverse effect if the adjacency graph is complete. Much stronger constraints can be derived from, for example, the label-compatibilities between pattern parts, or from pose information in the case of 3D object recognition. The usefulness of such information is, however, pattern dependent and considered beyond the scope of the present paper. In any case, for the simple patterns shown in Figure 3, and the low connectivity of the adjacency graphs of the montages, the relaxation method outlined here proved to be sufficient to obtain perfect part labeling. The results obtained using this technique are shown in Figure 3d.

4   3D Object Recognition using Range Data

4.1   Encoding of Object Surfaces

In the previous Section, we have illustrated the CRG method with a recognition problem involving 2D line patterns. For 3D recognition systems, the input can consist of intensity (brightness and/or color) data generated by a video camera, or of range (depth) data. The latter can be sensed by active vision (laser range finders or strip lighting devices) or can, for example, consist of sparse depth maps produced by Shape-from-X methods.

We deal with range data, and for the purpose of this paper, we do not deal with this initial sensing problem and simply assume that we already have view-dependent range (depth) maps of 3D objects. However, as in the 2D case, we deal with the recognition of isolated objects and objects in scenes. One of the main reasons for using such view-dependent data formats is that the computations of surface curvatures, or pixel labels in general, are restricted to what is visible. That is, there exists full view-independent surface information that is not visible: for example, the "inside regions" of some concave objects. The additional benefit of computing curvatures from such a data format (Monge patch data of the form $(x, y, z(x, y))$ is that more standard signal processing techniques can be used to regularize the evaluation of derivatives, etc. (see [17] for more details). What is important, however, is that we have computed object unary and binary part features with respect to the full 3D properties of the range data. That is, questions as to the benefits and deficits of view-dependent versus view-independent representations involves evaluations of both the data format and the types of features to be computed.

Full view-independent representations involve complete 3D descriptions of surface patches and the fact that these patch features are evaluated from view-dependent aspects is actually not the essential issue involved. For example, computing surface features that are invariant to rigid motions is as important to a "view-independent" representation as that of using full 3D CAD models. That is, for recognition purposes, it is the invariance of the representation that determines the degree of invariance in the models as much as the types of data inputs used. For these reasons, we have adhered to the view-dependent format. Further, the issue of the minimum number of views required to obtain correct identification of objects invariant to view is not so much a problem of the data formats but a problem of the types of object classes involved. For example, we only need one view of an ant and one of an elephant for fully invariant and correct 2-object classification performance!

Over the past decade, a variety of techniques have been developed for the registration of surface "shape" that produce representations which are invariant to rigid motions - a condition of central importance to robust Object Recognition Systems (ORS). Principal curvatures, Mean (H) and Gaussian (K) curvatures satisfy these conditions [18] though there are many different methods available for computing them. H and K are defined by:

$$H = \frac{1}{2} \frac{f_{xx} + f_{yy} + f_{xx}f_y^2 + f_{yy}f_x^2 - 2f_x f_y f_{xy}}{(1 + f_x^2 + f_y^2)^{\frac{3}{2}}} \qquad (6)$$

and

$$K = \frac{f_{xx}f_{yy} - f_{xy}^2}{(1 + f_x^2 + f_y^2)^2} \qquad (7)$$

for the Monge patch (view-dependent depth map) case where f_{uv} refers to partial differentiation of f with respect to u ($u = x$) and v ($v = y$) and $f(x, y)$ to the view-dependent range image.

Such computations require initial surface smoothing which is usually accomplished by fitting quadratic surfaces [19] or by low-pass filtering (surface blurring), after which partial derivatives are computed. Using this latter form of smoothing we have also used Fourier methods to compute the derivatives. That is, from the Differentiation Theorem [20] the partial derivatives of the function

60

$f(x, y)$ (representing, in this case the Monge patch surface model $(x, y, z = f(x, y))$) is determined (for each variable denoted by x) by:

$$\frac{\partial^n f}{\partial x^n} = F^{-1}((iu)^n F) \tag{8}$$

where F corresponds to the Fourier transform of f and F^{-1} to the inverse Fourier transform. That is, to partially differentiate an image $f(x, y)$ with respect to x, its Fourier transform is multiplied by the real (quadratic, second order of differentiation) or imaginary (linear, first order) ramp function $(iu)^n$ - resulting in even and odd bandpass filters. The benefits of such methods lie in the degree of "support" for computing $f_x, f_y, f_{xy}, f_{xx}, f_{xy}$ - the components of H and K. Furthermore, one of the main sources of "noise" in computing H and K lies in the division of images having different differential (bandpass) information - particularly in the regions of curvature zero-crossings. Our solution has been to compute zero-crossings, or segmentation, directly from the determinant of the Hessian ("shape" operator), the numerator of K):

$$S(x, y) = f_{xx} f_{yy} - f_{xy}^2 \tag{9}$$

segmenting the surface into convex, concave and planar regions. We then compute the complete H and K values within the resultant regions (see the following Section) using the low-pass filtering in conjunction with the spectral method for the computation of derivatives (see (8) above). The net result is to produce estimates of H and K with respect to a "scale" defined by the low-pass filter.

4.2 Segmentation

The issue of segmentation for ORS's, and for range data specifically, has received a good deal of attention in recent years. Common to most approaches is the development of surface part clustering in terms of similarities in surface point position, normals, or curvature information or surface curve fitting parameters. Segmentation, in these low-level terms does not guarantee the derivation of "parts" that are consistent with, for example, "model parts" defined by other processes, and some attempts have been made to split and merge such initially segmented regions, consistent with known patch feature bounds of the object parts in the database [7].

An alternative way of guaranteeing compatibility between model and test data parts is to use a segmentation procedure that is guaranteed to *apply equally* to both domains and uses features that are invariant to the parameterization of the surface. Fortunately, Mean (H) and Gaussian (K) curvatures satisfy these conditions. We have chosen to use zero-crossings of the determinant of the Hessian (see (9) above) as our segmentation procedure - which determines convex, concave and planar regions in a way which minimizes noise amplification that typically occurs when full H and/or K zero-crossings are evaluated. Such a segmentation procedure applies equally to models and data and is invariant to rigid motions. As mentioned above, we still use full H and K values to characterize each such region and so the initial segmentation is simply an adaptive data reduction method to package surface parts in ways that can be compared across data and models.

The major problem with using zero-crossings lies in determining what constitutes "zero". The problem of thresholds for zero-crossings has recently been

Predicate	Type	Computation
Unary	Size	U.D.1 Area
	Span	U.D.2 3D Spanning distance (Max)
		U.B.1 Perimeter
		U.B.2 mean Curvature
		U.B.3 mean torsion
Binary	B-type	B.B.1 length of jumps
		B.B.2 length of creases
	Jumpgap	B.D.1 bounding distance
		B.D.2 Centroid distance
		B.D.3 Maxdistance
	B-angle	B.A.1 differences in normal angles
		B.A.2 average bounding angle between surfaces
	N-angle	B.A.3 normal angle differences

Table 1: Typical Unary and Binary Surface Features

discussed [21]. Here, we have used a straight forward training approach where the threshold was determined from the maximum non-zero value of the Hessian's response (9) to the known planar background, assuming that scene objects are in front of a planar background [22].

4.3 Feature Extraction

In ORS, the purpose of segmentation is to enable an efficient data structure for the definition of models by the properties of surface patches and their relational features. Such features need to optimize two somewhat contradictory goals: invariance and uniqueness. The former refers to the need to represent models in ways which are invariant to rigid motions, pose, etc., while the latter refers to the development of representations which uniquely define the model. For example, H and K are invariant to rigid motions but are only unique up to the general type of surface and do not uniquely define it. Such uniqueness comes from the Gauss-Weingarten equations with the Mainardi-Codazzi compatibility equations defining the constraints on the differential (tensor) operators [18].

Model surface features are usually of two generic forms [22]. Unary features refer typically to (local) surface patch properties (such as curvatures), global patch properties (such as areas), or to to properties of patch boundaries (such as perimeter). Binary features typically capture part relationships such as distances, angles, and also include boundary relationships (see below). Typical examples of all feature types are shown in Table 1 (center column) and those used in this implementation are shown in Table 1 (right column). The right-hand column groups features into different types, unary curvatures (U.C), unary distance (U.D), unary boundary (U.B) and binary boundary (B.B), binary distance (B.D) and binary angle (B.A).

We have employed statistics of the pixel ("local") Mean and Gaussian curvatures of each patch. These features define surface shape characteristics that are invariant to rigid motions. Such measures eliminate the need for less quantitat-

ive features, such as "sense" which defines only the type of surface shape. Such "local" unary features are view-independent and they enable the identification of a part when only a section of it is visible - given that the section is representative of the part shape (i.e., if it is an unbiased sample). "Global" unary features are less invariant since they are computed over a full patch and so are subject to, for example, self-occlusion for different views. We have, however, included the areas, perimeters and spanning distances as already used in current implement-ations - though area is directly related to the average of the Gaussian curvature. We have defined part boundary (edge) properties by their associated curvature and torsion statistics. In particular, the torsion statistic defines the degree to which the bounding contours deviate from planarity and, from the Serret-Frenet equations, these features uniquely define a contour in 3D - invariant to rigid mo-tions. Here we have computed boundary contour curvatures(κ) and torsion(τ) statistics by their finite different forms. Curvature is defined by

$$\kappa(s) = \frac{1}{\mid t_s(s) \mid}$$ (10)

where

$$t_s(s) = (x_s(s+1) - x_s(s), y_s(s+1) - y_s(s), z_s(s+1) - z_s(s))$$

and

$$X_s(s) = (x(s+1) - x(s), y(s+1) - y(s), z(s+1) - z(s))$$

for s being the parameter of the curve (contour) equation

$$X(s) \equiv (x(s), y(s), z(s)).$$

Torsion is defined by

$$\tau(s) = -b_s(s) \cdot n(s)$$ (11)

where

$$b(s) = \tau(s) \times n(s)$$

with $\times$ corresponding to vector (cross) product, and

$$n(s) = k(s)/ \mid b(s) \mid$$

corresponding to the normal to the curve at s.

We have used the binary features total lengths of jumps and creases between shared contours, consistent with recent object recognition systems (for example, [5]). Such continuous versions of these binary features are more suitable for a fea-ture space (vector space) representation and for the rule generation (clustering) procedure proposed in the current paper.

4.4 Learning Structural Descriptions of Objects

In the 3D object recognition example, seven synthetic objects were learned at training time. Each object was presented in isolation and from 20 different views (equally spaced over three Euler angles). Examples views of each object are shown in Figure 4.
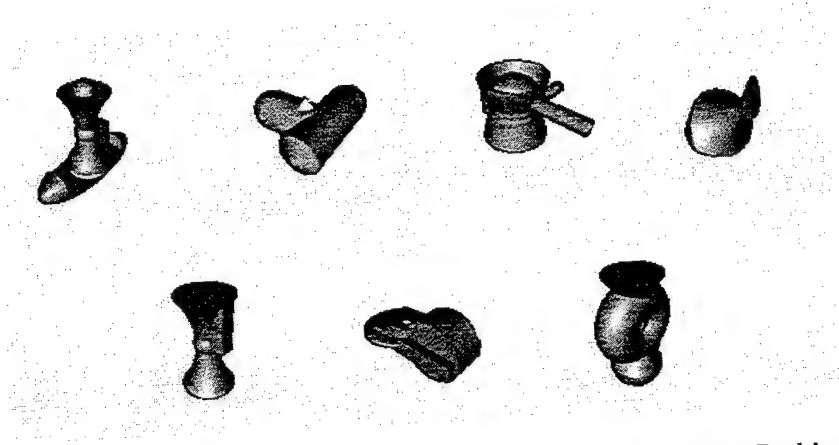
Figure 4: Rendered view of each of the seven objects used in the 3D object recognition experiment.

Analysis of the depth maps for each object and view proceeded as described in the Sections 4.1 - 4.3, resulting, for each view, in a set of depth map regions that were described by the unary and binary features shown in Table 1.

These rules were then used to classify montages of objects such as shown in Figure 5. Here, the top row shows the depth maps of two montages, the middle row shows the segmented depth map regions, and the bottom row shows the region classifications. For the montage on the left, object overlap is relatively small, and for the montage on the right it is substantial. Chain analysis and part classification proceeded as described in Section 3, both for the WTA-scheme and the RL scheme. In the RL scheme, relaxation was run for 20 iterations, followed by a WTA iteration on the final evidence vectors to produce a unique classification for each region. A summary of the correct region classification, for both schemes and for both the left and right montage in Figure 5, is given in Table 2. From the results in this Table, as well as from the classification map in Figure 5, it is clear the the rules produced by CRG are capable of classifying correctly a majority of object regions.

5   3D Object Recognition using Intensity Data

The **blocks** example presented in this Section consists of various configurations of colored toy blocks. The configurations are learned in isolation (see Figure 6) and have to be identified in more complex arrangements (see Figure 7). The training set consisted of 5 classes of block configurations, each with three training examples, and the test arrangements consisted of up to 20 blocks.

Images of the training and test scenes were captured with a color camera. Preprocessing was fairly simple, consisting of a *segmentation* stage and a *feature extraction* stage. Segmentation was achieved using a form of K-means clustering (minimizing within-cluster variance in feature space) on position (x, y) and color (r, g, b) attributes [23]. For the resulting clusters, small clusters were merged with larger neighbor clusters in order to eliminate spurious image regions. Given the rich image information, it is not surprising that the resulting image regions
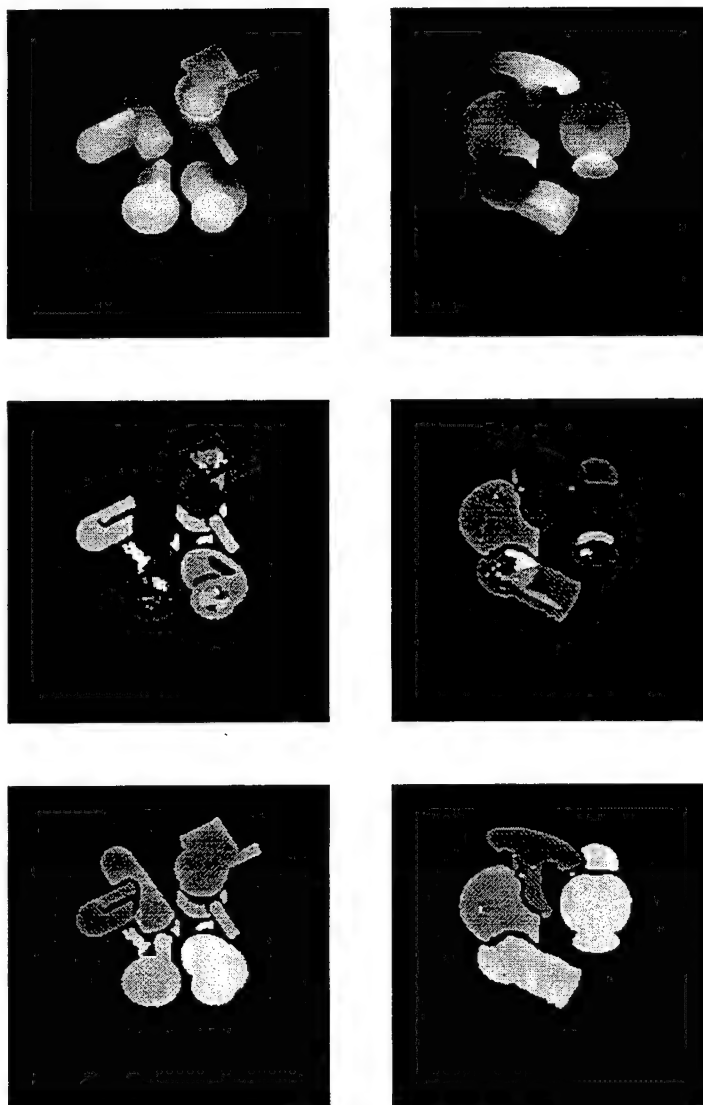
Figure 5: Two different montages of synthetic objects (left and right panel). (Top) Range images of two scenes used to test object identification and classification. (Middle) Segmented depth map regions (defined by different grey levels) from the zero-crossings of Gaussian curvature. (Bottom) Region classification for the two montages. Different grey levels define different class labels.

	Number of parts	WTA scheme	RL scheme
left scene	82	76	78
right scene	63	53	56

Table 2: Number of correct region classifications for two scenes in Figure 5, using the WTA-scheme and the RL-scheme.

correspond fairly well to the individual blocks.

In the feature extraction stage the following unary features were extracted for each image region: size (in pixels), compactness ($perimeter^2/area$), and the normalized color signals $R/(R+G+B)$, $G/(R+G+B)$, and $B/(R+G+B)$. For pairs of image regions the following binary features were computed: absolute distance of region centers, minimum distance between the regions, distance of region centers normalized by the sum of the region areas, and length of shared boundaries normalized by total boundary length.

For the training data, CRG analyzed 276 different paths of pattern parts and produced 32 rules: 9 U-rules, 4 UB-rules, 12 UBU-rules, 3 $UBUB$-rules, and 4 $UBUBU$-rules. From the distribution of rule types, it is evident that CRG used predominantly unary features for classification. Given the fact that CRG has a strong tendency to produce shallow cluster trees and short rules (see Section 2.2), and given the fact that the unary features are quite diagnostic (see Figure 6), this result is not surprising. However, each unary and binary feature was used in at least some of the classification rules.

Classification performance was tested with several complex configurations of block patterns, two of which are shown in Figure 7, together with the classification results. Classification proceeded as described in Section 3, using the chain analysis and relaxation labeling solution. For both scenes, all parts (11 in Figure 7a, 17 in Figure 7b) were classified correctly with the exception of a single part from the class-4 configuration (see Figures 7c and 7d).

For comparison purposes, we have analyzed the block example using classical decision trees [24]. In the first analysis, each image part P of the training and test images was described by 13 features. These features consisted of the five unary features of P (see above), the four binary features (see above) of the relation between P and its closest neighbor, and another four binary features of the relation between P and its second-closest neighbor. For the class-1 cases which consisted of two parts only, the feature values for the second binary relation were set to "unknown". A decision tree was generated using C4.5 with default parameters [24], and the resulting tree was used to classify all parts of the test scenes in Figure 7. In each of the two scenes, 3 parts were misclassified. The good performance obtained with C4.5 is consistent with the observation that the use of higher-order relational information does not seem to be crucial for successful classification of this data set.

In this first analysis, features of all UBB-triples (unary features and binary features of relations with two other parts) were used for classification. A second analysis, using UBU-triples (with 14 features: the same five unary features of all pairs of parts, as well as the same four binary features of their relation) was performed, but the results cannot be interpreted as easily. For the scene
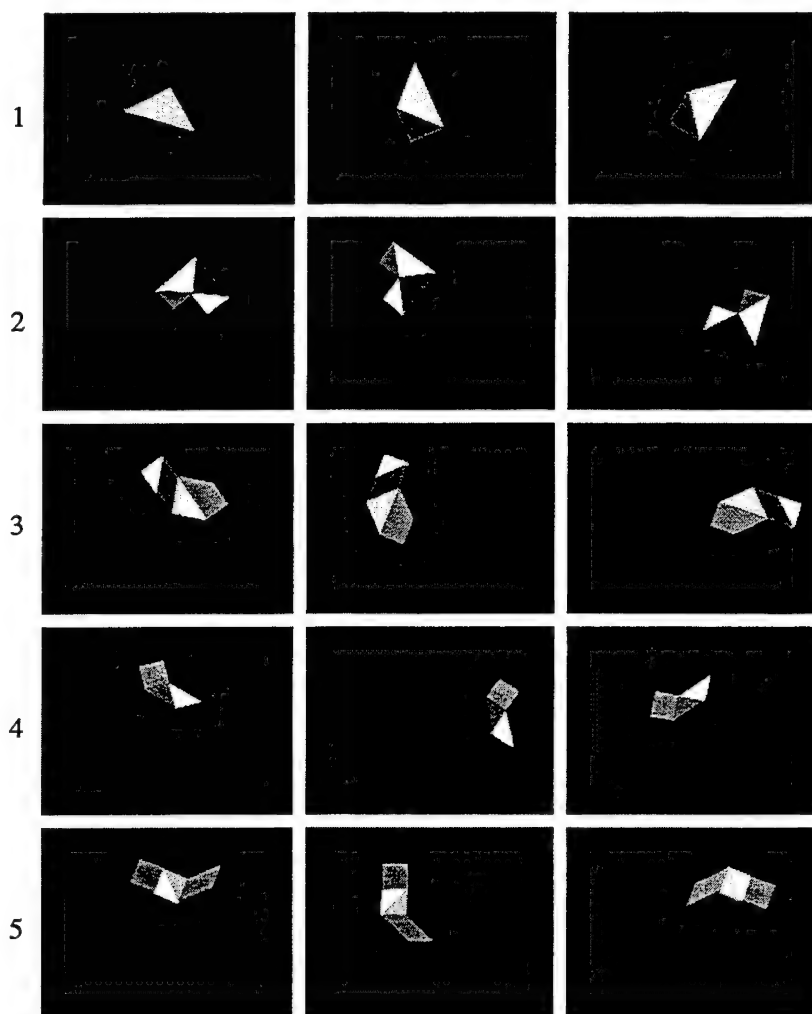
Figure 6: Images of five classes of toy block configurations with three views each. The image parts are described by the unary features size, eccentricity and the three normalized color coordinates. Pairs of image parts are described by the binary features of midpoint distance, area-normalized midpoint distance, minimum distance and normalized shared boundary length.

in Figure 7a, 33 out of 110 *UBU*-triples or 30% were misclassified, and for the scene in Figure 7b 103 out of 272 *UBU*-triples or 37.8% were misclassified. One reason for the error rate being so high is the fact that no analysis corresponding to the chain analysis described in Section 3 was performed with the C4.5 results. However, the error rates seem to be too high to be corrected using the relaxation scheme proposed there.

A general point is, however, more important. The CRG method generates rules of (minimal) variable length *optimized for a given training set*, whereas the decision tree (C4.5) *fixes* the dimensionality of the feature space and rule length.
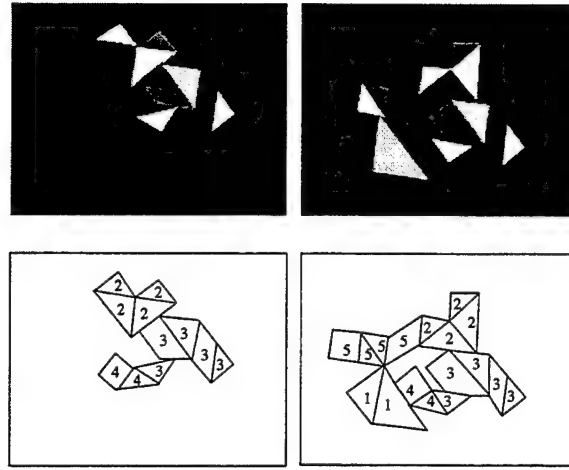
Figure 7: Two block scenes and their classifications. (a) Block scene consisting of 11 blocks corresponding to examples of classes 2, 3, and 4. (b) Block scene consisting of 17 blocks corresponding to examples of all classes. (c) Classification result for block scene in (a) with region labels corresponding to classes. (d) Classification result for block scene in (b) with region labels corresponding to classes.

The choice of *UBB*-triples for the block example lead to a C4.5 performance that was essentially the same as that of CRG, but for the *UBU*-triples C4.5 performance was much worse. This choice has to be done *a priori* whereas it is adjusted dynamically in the CRG method.

6 Discussion

CRG develops structural descriptions of patterns in the form of decision trees on attribute bounds of ordered predicates (see Figure 2). It is thus useful to compare it with other techniques from Machine Learning which attain similar ends symbolically.

CRG shares with ID3 / C4.5 [25, 24], and related techniques, similar methods for the search and expansion of decision trees. However, these latter techniques were not designed to generate rules satisfying label compatibility between unary and binary predicates. CRG, on the other hand, is explicitly designed to develop rules for unique identification of classes with respect to their "structural" (i.e. linked unary and binary feature) representation. The application of C4.5 to the block example in the previous Section was therefore somewhat misleading, in the sense that label-compatible data were generated beforehand.

In decision trees, features or attributes are analyzed within a single feature space, independent of their relationships or arities, and no preferential order is imposed on the features. In contrast, the CRG method generates conditional features spaces as required, and it defines a preferential ordering on attributes in the sense that, for example, a split of a *U*-feature is preferred over a split of *UBU*-features. This preferential order leads to the generation of shallow cluster

trees and short rules, as discussed in the previous Sections.

Decision trees operate on a fixed path length (for example, the *UBB*- or *UBU*-triples in the block example) and thus *force*, a priori, the choice of relational structures to be analyzed. CRG, on the other hand, has variable length path expansion determined by the number of parts and their relations that are required to uniquely define patterns. Consequently, CRG is superior to classic decision trees when classification relies on relational information and does so to different degrees for different examples or classes. Under these circumstances one would be forced to use high-dimensional features spaces with classical decision trees, whereas CRG would generate minimal depth trees. Furthermore, generating minimum depth trees is of crucial importance since the number of paths grows exponentially with path length.

In summary on can say the classical decision trees are *attribute-indexed* in the sense that various levels in the tree define different attributes and the nodes define different attribute states. To this decision tree structure, CRG adds another layer, a *part-indexed* tree of features spaces, each with its own attribute-indexed decision tree. With this tree of decision trees, CRG imposes both a limit on the number of attributes that are being considered, and an ordering on the evaluation of attributes.

CRG uses linearly separable attribute bounds for rules or generalizations. Since CRG is part-indexed and not explicitly attribute-indexed, this is not required but has been used in this implementation for comparison purposes. Finally, the computational complexity of CRG is, in principle, identical to decision trees insofar as the attribute testing and splitting procedures are similar. However, the unique *relational aspects* of CRG may or may not result in more efficient learning, depending on the type of learning context.

Recently, Quinlan [26] and Muggleton and Buntine [27] have investigated general methods for learning *symbolic* relational structures in the form of Horn clauses in the following sense. In FOIL, [26] considers the problem of learning, from positive examples (closed world) or positive and negative examples, conjunctions of literals that satisfy

$$C \leftarrow L_1, ..., L_m$$

where C would correspond, in our case, to a class label. FOIL solves such problems by expanding the literals - adding predicates and their variables - to the right-hand-side to maximize the covering of positive instances and to minimize inclusion of negative ones. In this framework, then, CRG is also concerned with generating similar class descriptions of the specific forms:

$$
\begin{aligned}
C_1^1 &\leftarrow U^1(X), B^1(X,Y), U^2(Y), B^2(Y,Z), U^3(Z), ... \\
&... \\
C_1^{n_1} &\leftarrow U^1(X), B^1(X,Y), U^2(Y), B^2(Y,Z), U^3(Z), ... \\
&... \\
C_m^1 &\leftarrow U^1(X), B^1(X,Y), U^2(Y), B^2(Y,Z), U^3(Z), ... \\
&... \\
C_m^{n_m} &\leftarrow U^1(X), B^1(X,Y), U^2(Y), B^2(Y,Z), U^3(Z), ...
\end{aligned}
$$

However, CRG differs significantly from FOIL in the following ways:
1) the choice of unary U-rules and binary B-rules as bounded attribute (feature) states, is determined within continuous unary and binary feature spaces;

2) the ordering of literals must be *satisfied* in the rule generation;

3) the search technique uses backtracking and recursive splitting and

4) the resultant rules are not only Horn clauses but each literal *indexes* bounded regions in the associated feature space (as shown in Figure 2).

The CRG method is an example of the general solution to complex pattern recognition problems involving the generation of rules, as bounded predicate Horn Clauses, which are linked together in ways that determine "structure" uniquely enough to identify classes but enable generalization to tolerate distortions. Both aims, uniqueness and generalization, are not explicitly guaranteed in other methods, such as neural networks or decision trees. Further, uniqueness and generalization constitute the equivalent of a "cost" function in CRG, and the search technique has been developed to satisfy these constraints.

Finally, CRG raises the question as to what really is a "structural description" of a pattern. CRG simply generates conditional rules that combine an attempt to generalize the pattern definitions in terms of feature bounds and to restrict the description lengths as much as possible. For complex and highly variable training patterns, CRG can generate a large number of rules which can be thought of as a set of *equivalent descriptions* of the pattern structure. It is possible to determine the more frequently occurring paths and associated feature bounds from the cluster tree, if the notion of "commonness" is deemed necessary for a structural description. However, this may not really be a meaningful definition of structure. Rather than producing a singular rule structure, a "structural description" is defined by a *set of rules* that CRG generates from a set of training patterns.

CRG offers a way for automatically generating structural descriptions which enable rapid tree-based search techniques in complex scene data. For this reason it provides a most useful approach to problems in target detection, surveillance and security applications where not all objects in the scene are required to be identified but those which are also require robust description and rapid detection.

Acknowledgments

References

[1] R. E. Tarjan and A. E. Trojanowski, "Finding a maximum independent set," *SIAM Journal of Computing*, vol. 6, pp. 537–546, 1977.

[2] D. Ballard and C. Brown, *Computer Vision.* Englewood Cliffs, NJ: Prentice-Hall, 1982.

[3] W. E. L. Grimson, *Object Recognition by Computer.* Cambridge, MA: MIT Press, 1990.

[4] P. Flynn and A. K. Jain, "3D object recognition using invariant feature indexing of interpretation tables," *Computer Vision, Graphics, and Image Processing*, vol. 55, pp. 119–129, 1992.

[5] A. K. Jain and D. Hoffman, "Evidence-based recognition of objects," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, pp. 783–802, 1988.

[6] T. Caelli and A. Pennington, "An improved rule generation method for evidence-based classification systems," *Pattern Recognition*, vol. 26, pp. 733–740, 1993.

[7] P. Flynn and A. K. Jain, "Three-dimensional object recognition," in *Handbook of Pattern Recognition and Image Processing, Volume 2: Computer Vision* (T. Y. Young, ed.), New York, NY: Academic Press, 1993.

[8] K. Ikeuchi and T. Kanade, "Automatic generation of object recognition programs," *Proceeding of the IEEE*, vol. 76, pp. 1016–1035, 1988.

[9] R. C. Bolles and P. Horaud, "A three-dimensional part orientation system," *The International Journal of Robotics Research*, vol. 5, pp. 3–26, 1986.

[10] M. A. Fischler and R. A. Elschlager, "The representation and matching of pictorial structures," *IEEE Transactions on Computing*, vol. 22, pp. 67–92, 1973.

[11] R. Mohan and R. Nevatia, "Using perceptual organization to extract 3-d structures," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, pp. 1121–1139, 1989.

[12] D. G. Lowe, "Three-dimensional object recognition from single two-dimensional images," *Artificial Intelligence*, vol. 31, pp. 355–395, 1987.

[13] R. Michalski and R. E. Stepp, "Automated construction of classifications: Conceptual clustering versus numerical taxonomy," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 5, pp. 396–409, 1983.

[14] K. C. C. Chan, A. K. Wong, and D. K. Y. Chiu, "Learning sequential patterns for probabilistic inductive prediction," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 24, pp. 1532–1547, 1994.

[15] W. F. Bischof and T. Caelli, "Learning structural descriptions of patterns: A new technique for conditional clustering and rule generation," *Pattern Recognition*, vol. 27, pp. 1231–1248, 1994.

[16] U. Fayyad and K. Irani, "On the handling of continuous-valued attributes in decision tree generation," *Machine Learning*, vol. 8, pp. 87–102, 1992.

[17] E. Barth, T. Caelli, and C. Zetsche, "Image encoding, labelling and reconstruction from differential geometry," *Computer Vision, Graphics and Image Processing*, vol. 55, pp. 428–446, 1993.

[18] M. D. Carmo, *Differential geometry of Curves and Surfaces*. Englewood Cliffs, NJ: Prentice Hall, 1976.

[19] P. Besl and R. Jain, "Invariant surface characteristics for 3D object recognition in range images," *Computer Vision, Graphics and Image Processing*, vol. 33, pp. 33–80, 1986.

[20] A. Rosenfeld and A. Kak, *Digital Picture Processing.* New York, NY: Academic Press, 1982.

[21] N. Yokoya and M. Levine, "Range image segmentation based on differential geometry: A hybrid approach," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, pp. 643–649, 1989.

[22] T. Caelli and A. Dreier, "Variations on the evidenced-based object recognition theme," *Pattern Recognition*, vol. 27, pp. 185–204, 1994.

[23] A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data.* Englewood Cliffs, NJ: Prentice Hall, 1988.

[24] J. R. Quinlan, *C4.5 Programs for Machine Learning.* San Mateo, CA: Morgan Kaufmann, 1993.

[25] J. R. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, pp. 81–106, 1986.

[26] J. R. Quinlan, "Learning logical definitions from relations," *Machine Learning*, vol. 5, pp. 239–266, 1990.

[27] S. Muggleton and W. Buntine, "Machine invention of first-order predicates by inverting resolution," in *Proceedings of the Fifth International Conference on Machine Learning*, pp. 339–352, 1988.

[28] A. Pearce, T. Caelli, and W. F. Bischof, "Rulegraphs for graph matching in pattern recognition," *Pattern Recognition*, vol. 27, pp. 1231–1248, 1994.

Interpretation of Dynamic Interaction in Image Sequences

Sandy Dance and Zhi-Qiang Liu

Department of Computer Science

University of Melbourne

Parkville, 3052, Australia

`sandy@cs.mu.OZ.AU`

Abstract

In this paper we discuss the problems and opportunities involved in dynamic scene analysis, especially with regard to complex continuous situations, such as tactical analysis of ground traffic, air traffic, and ships in convoy, dealing with these modalities in surveillance, civilian management and battlefield management. We describe the fundamental issues in dynamic systems, and discuss some approaches in the literature. We then describe some of the problems in various traffic contexts, and present a hierarchical multi-agent architecture that is suited to symbolically analyse real-time dynamic processes, which is also capable of inferring intentional behaviour at a high level.

1 Introduction

In tactical analyses of continuous scenes of ground traffic, air traffic, and ships in convoy, in surveillance, civilian management and battlefield management, it is essential to work at the *intensional* level in order to make meaningful or useful predictions or descriptions. This is simply because the entities are under the control of humans who themselves reason at this high level. This presents a major challenge to designers of automated systems that perform such analyses.

As Van Gelder[18] has pointed out, there are two main approaches to cognition in continuous temporal contexts, the dynamic (or connectionist) and the computational (Figure 1). The former approach is exemplified by neural nets and Kosko's[9] fuzzy cognitive maps, in which nodes (objects, agents, processes or modules) are connected together by arcs (links, synapses, relationships, channels) which convey continuous time-varying real values. The nature of the cognition is determined by the network structure, and the object structure. For instance, in neural nets the incoming weighted link values are summed, normalised with the sigmoidal function and output to other links. In the computational approach, as articulated in Newell and Simon's "physical symbol system hypothesis"[13], the links between modules carry not time-varying values but symbols, hence the actions of the modules are batch-oriented rather than a continuous reaction to the input values.
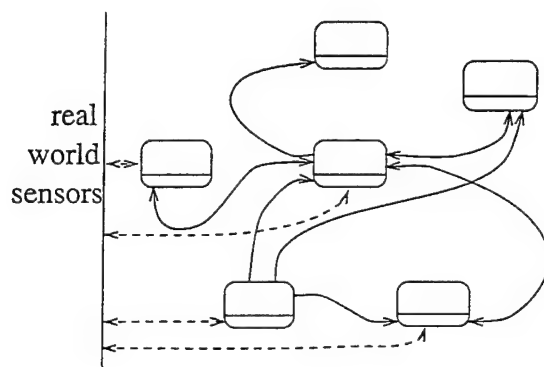

Figure 1: *The agent-oriented concept: agents communicate with other agents via the links. Some agents communicate with the real world sensory data. In the connectionist approach the links convey time-varying real values only, and in the computational approach the links convey messages containing symbols and/or values.*

2   Previous Approaches

In the connectionist camp is the work of Huang, Koller et al [8] who use a Bayesian belief network and inference engine (HUGIN[1]) in sequences of

highway traffic scenes to produce high-level concepts like "car changing lane" and "car stalled". This approach is not dynamic as the network is rolled forward frame by frame rather than continously updated. In general, belief networks propagate values around the network as vectors, with each link having associated matrices reflecting the conditional probabilities [14]. One problem with the Bayesian inference is that each node must have a set of *exhaustive* and *mutually exclusive* states, and the *a priori* conditional probabilities, which are often difficult to obtain in real vision applications.

A purely symbolic approach is exemplified by the work of [2, 12, 16] in which dynamic image sequences of traffic scenes or soccer games are analysed using event recognisers. Here at each instant the geometric relationships are described by a geometric scene description (GSD), which is updated frame by frame. This is input into the event recogniser, which is variously, a transition network[2, 16] (which works like a parser in linguistics working on a stream of tokens), or a set of logical clauses[12] with control based on unification and backtracking, ie, the same control structure as Prolog. We feel that this symbolic approach does not take full advantage of the constraints offered by dynamics, and it requires a lot of computer resources having to roll forward the GSD from frame to frame, nor is this work based on an agent-oriented approach such as the one we espouse.

Work which is an amalgam of the connectionist and symbolic approaches is that of [15] in which a behaviour net (similar to a spreading activation network) conveys an "energy" value between modules. However, needing to convey some symbolic information, they use "pronomes" – places where symbols are stored and accessed by other modules. Thus in this system, any symbolic information is either conveyed as dedicated channels between modules, or is stored in special modules which are updated through another set of dedicated channels, one channel or module for each possible value of all the symbols. It becomes apparent that this system scales badly as the domain complexity increases.

3   Situatedness and Intentionality

In recent years, it has become clear that computer systems dealing with (modeling) dynamic real situations need to be "embedded" in that situation, that is, interacting with their environment, and in fact such in-

teraction, in the case of computational systems, "grounds" or provides meaning to the symbols used. This is particularly true of systems for processing spatial relationships, for instance, the analysis of vehicle interactions. Situationists have made radical claims that cognitive systems use *only* context for their representations, and that there are no internal representations [5]. However, Slezak [17] has pointed out the need to distinguish the representation used internally to implement cognitive systems from that used for external communication, and shows how this clarifies the situationists' claims.

Cognitive systems for processing complex dynamic spatial information need to perform at the intentional or semantic level, for instance, it is clear that understanding the movements of traffic under the control of humans is not going to successfully predict behaviour without in some way modeling the intentional states of the drivers. Specifically, in ground traffic, to understand why a car is slowing down, it is necessary to model the give-way traffic rules which indicate that the car must give-way to another, and the driver's intention to adhere to that rule. This can be generalised to any complex dynamic system involving "rational" agents, where each agent runs models of any other agent they are interacting with[19].

4 A Dynamic Symbolic Interpreter

In this section, we describe one such embedded system which models the intentional level with a symbolic network. The dynamic approach is not currently viable due to the lack of a high-level description language and top-down construction paradigm. We employ a computational based system which emulates a dynamic system (in the sense of van Gelder[18]). Here agents actively forward messages through the network provided the change from the previous state is over a threshold. Thus the symbolic system reacts to any changes while avoiding performing the same calculation repeatedly. Agents dealing with spatial concepts also have a dynamic aspect in that they interact with real-world sensors or an active spatial database.

This system is built upon our previous work [6] in which short sequences (usually 3 frames) of traffic images are interpreted according to the road rules - reflecting driver intentionality, dynamical constraints, and data fusion from video input and the traffic light controller (see Fig-

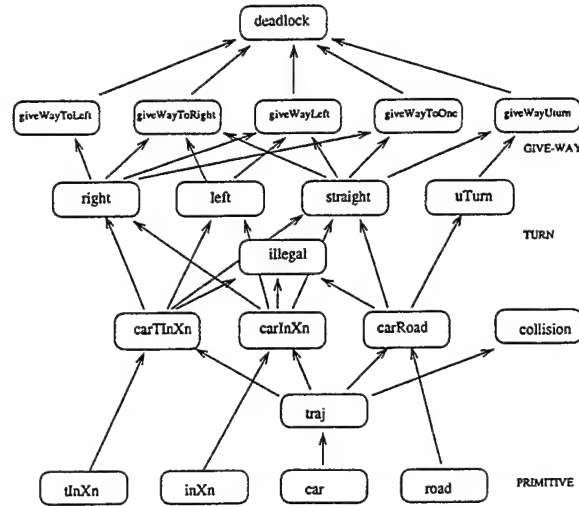ure 2). We explore the general requirements for a *dynamic* high-level interpreter in a changing world.

Figure 2: *The static traffic scenario network-of-agents. The arrows refer to activation messages, inquiry and update messages are not shown. inXn refers to "intersection", tInXn to "T-intersection", carInXn to the concept of a car in an intersection, carTInXn refers to a car in a T-intersection, and carRoad to a car in a road.*

In our dynamic network, we can distinguish between a number of agent types:

- Instantaneous: an agent dealing with concepts concerned with a single instant (frame), ie, dealing with a particular segment in an image. These agents have a lifetime of a few frames only.

- Continuous: an agent dealing with a concept that has continuously time-varying parameters. For instance, a car with parameters position and velocity. This agent is constantly updated with the current values for the object involved, and is removed when the object disappears. When any change in its parameters occurs (over a given threshold), this agent sends messages to other relevant agents in the network.

- Event: an agent that deals with an event in space-time (ie, a vehicle coming to a stop), which would carry as a parameter the time of the event. Such agents would have lifetime parameters as well, and upon expiry, would disappear.

- Durational: an agent dealing with extended events, for instance the entire turn sequence of a car. These agents would carry the start and stop times of the event.

These agents can deal with single objects in the scene, or more complex concepts based on relationships or interactions between objects.

Output from this system derives from top-level scenario agents which tell the story of an interaction between two or more objects in the scene, ie, an event involving cars approaching each other, realising they are in a give-way relationship, one car slowing, then stopping, and eventual disappearance of the cars from the scene. As well, the system provides a short term story generator that is able to tell the instantaneous picture upon the user's request.

5   Uncertainty

In this system, dealing with the problems of data fusion, uncertainty handling is essential, as data is coming from simultaneous disparate sources, not only from visual agents, but potentially from other sensory modalities like traffic light controllers, magnetic road sensors, radar, sonar, GPS, etc. If the sensor's output becomes unavailable, the system must use a form of default reasoning based on expert judgment. An approach to handling uncertainty in this kind of environment is pursued in [11], where sensory data are regarded as values in a frame of discernment which provides evidence for predicates with fuzzy membership values. The approach also incorporates two new elements, a binary variable reflecting the availability of the sensor, and a belief function (a la *Dempster-Shafer*[7, 10]) which reflects expert opinion, and is used as the default value in the case of no sensory data. These three elements are combined in a belief measure "$g(v, \mu(x), Bel(x))$" which has the desired intuitive properties.

6 Scenarios

The system described above can be implemented in a number of possible scenarios. The first is that of ground traffic, which has been chosen for the clarity and the familiarity of the domain and its concepts, and because an appropriate set of rules exists. This extends the work done earlier for "instantaneous" (ie, 3 video frames) views. To this end, we will be using a 3D motion detection and classification method developed by Bruton et al [3, 4] who's adaptive 3D recursive filters efficiently generate trajectories from video sequences (see Figure3). Output from this scenario will consist of high-level descriptions of driver expectations and intentions, and subsequent vehicle interactions.
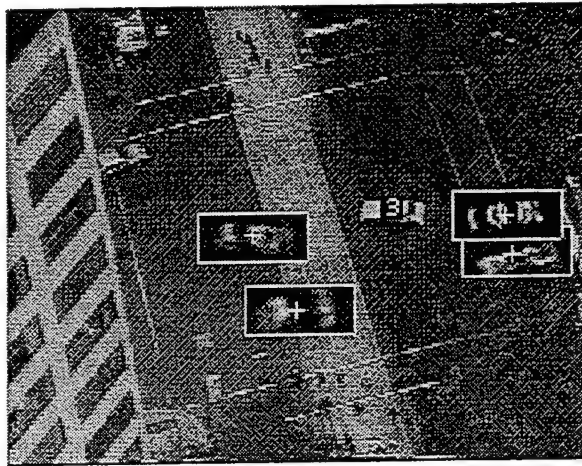

Figure 3: *Intermediate output for a single frame from Bruton et al's 3D motion detection and classification method, the rectangles are areas picked out by the motion detector, the numbered car has been classified as a right-turner.*

Other two dimensional domains which are potential domains for implementation are those of shipping in port or combat, and troop movements. For each of these there are high-level intentional concepts involved, together with dynamic, spatial interactions.

The air traffic control scenario is conceptually similar to the ground traffic control, but includes the added complexity of a third spatial di-

mension. In this scenario, pilot-to-pilot interactions are usually mediated through the air traffic controller, but are nonetheless real. Having a means of estimating the pilot intentional level allows the interesting possibility of a proximity alarm based on expectations of where the pilot intends the aircraft to be in the future.

This technique is also useful in air combat analysis, where the pilot-to-pilot interactions are more direct, based on the spatial relationships of the craft as perceived by the pilots. Again, the intentional level, as derived from the aircraft time-spatial coordinates based on either radar or onboard inertial guidance output, provides the possibility of deep analysis of the aircraft interactions.

One consideration in this system is how it scales from a few dozen entities to several thousand. On the face of it, it should scale as the number of possible interactions between entities, ie, about the square of the cardinality (ie, $\mathcal{O}(n^2)$). However, with judicious choice of concepts, together with conceptual clustering of entities into groups forming entities of a higher level, the system should scale at about the rate of the number of nodes in a hierarchical tree, ie, about $\mathcal{O}(n \log n)$, which would be acceptable.

It should be noted that as it stands, the system and all its concepts is hand built, meaning it has no automatic or machine learning algorithms. Thus, what would be useful is a front-end to allow easy construction of the system. Ultimately, we would like a system that when exposed to its environment learns its own concepts and links them together appropriately.

7 Conclusion

In this paper we have outlined an approach to dealing with dynamic systems of interacting entities using a high-level network-of-agents symbolically dealing with intentional states. This work builds upon previous work focused on interpretation of static scenes, but is extended to deal with dynamic interactions between entities and the world.

It is suggested that such an appraoch would be useful for interpreting the intentional states of aircraft pilots and ground vehicle drivers, especially with regard to their interactions with other vehicles in both civilian and battlefield domains.

References

[1] S. K. Andersen, K. G. Olesen, F. V. Jensen, and F. Jensen, HUGIN – a shell for building Bayesian belief universes for expert systems, in *IJCAI-89*, Detroit, 1080–1085 (1989).

[2] E. Andre, G. Herzog, and T. Rist, On the simultaneous interpretation of real world image sequences and their natural language descriptions: the system SOCCER., in Y. Kodratoff, editor, *ECAI 88. Proceedings of the 8th european conference on artificial intelligence*, UK, 449–54 (1988).

[3] L. T. Bruton and N. R. Bartley, The enhancement and tracking of moving objects in digital images using adaptive three-dimensional recursive filters, *IEEE Transactions on circuits and systems*, **33**(6), 604–612 (1986).

[4] L. T. Bruton, N. R. Bartley, and Z. Q. Liu, On the classification of moving objects in image sequences using 3D adaptive recursive tracking filters and neural networks (in press), in *Proc. IEEE International Conference on Neural Networks*, Perth, Australia (1995), (invited paper).

[5] W. J. Clancey, Situated cognition: How representations are created and and given meaning, in *AERA 1991 Symposium*, Chicago (1991).

[6] S. Dance, T. Caelli, and Z.-Q. Liu. *Picture Interpretation: a Symbolic Approach*, Series in Machine Perception and Artificial Intelligence. World Scientific Publishing Co., Singapore. (1995).

[7] A. P. Dempster, A generalization of Bayesian inference, *Journal of the Royal Statistical Society*, **30**, 205–247 (1968).

[8] T. T. Huang, D. Koller, J. Malik, G. Ogasawara, B. Rao, S. Russell, and J. Weber, Automatic symbolic traffic scene analysis using belief networks, in *Proc of AAAI-94*, Seattle (1994).

[9] B. Kosko. *Neural Networks and Fuzzy Systems*. Prentice Hall (1992).

[10] J. D. Lowrance, T. M. Strat, L. P. Wesley, T. D. Garvey, E. H. Ruspini, and D. E. Wilkins, The theory, implementation and practice of evidential reasoning, SRI Project 5701, SRI International, Menlo Park, CA 94025 (1991), Final report.

[11] A. Mogre, R. McLaren, J. Keller, and R. Krishnapuram, Uncertainty management for rule-based systems with applications to image analysis, *Transactions on Systems, Man, and Cybernetics*, **24**(3), 470–481 (1994).

[12] B. Neumann, Natural language description of time-varying scenes, in D. L. Waltz, editor, *Semantic Structures: advances in natural language processing*, Lawrence Erlbaum, Hillsdale, N.J, 167–207 (1989).

[13] A. Newell and H. Simon, Computer science as empirical enquiry: Symbols and search, *Communications of the Association for Computing Machinery*, **19**, 113–126 (1976).

[14] J. Pearl. *Probabilistic reasoning in intelligent systems: Networks of plausible inference*. Morgan Kaufman, San Mateo, CA (1988).

[15] B. Rhodes, Pronomes in behavior nets, in *Cognitive Science Society Conference '95* (1995).

[16] J. R. J. Schirra, G. Bosch, C. K. Sung, and G. Zimmermann, From image sequences to natural language: a first step toward automatic perception and description of motions, *Applied Artificial Intelligence*, **1**(4), 287–305 (87).

[17] P. Slezak, Situated cognition: Empirical issue, paradigm shift or conceptual confusion, in A. Ram and K. Eiselt, editors, *Proceedings of the Sixteenth Conference of the Cognitive Science Society*, Atlanta, Lawrence Erlbaum, Hillsdale, New Jersey, 806–811 (1994).

[18] T. van Gelder and R. Port, It's about time: An overview of the dynamical approach to cognition, in R. Port and T. van Gelder, editors, *Mind as motion: explorations in the dynamics of cognition*, MIT Press, Cambridge MA, 1–41 (1995).

[19] M. Wooldridge and N. R. Jennings. *Intelligent Agents: Theories, Architectures, and Languages*, **890** of *Lecture Notes in AI*. Springer-Verlag (1995).

Integrating and Managing Multimodal Information Sources: A Problem for Defence

Marilyn CROSS†[1]
Christian MATTHIESSEN††

† Senior Research Scientist
Defence Science and Technology Organisation
DSTO - C3 (Fernhill)
PO BOX E33
Queen Victoria Terrace
Canberra ACT 2600
Australia

Phone: +61 6 265 8026
Fax: +61 6 265 8010
Email: mcross@itd.dsto.gov.au

†† Associate Professor
School of English and Linguistics
Macquarie University
NORTH RYDE 2109
Phone: +61 2 850 7756
Email: cmatthie@pip.engl.mq.edu.au

1.   Managing Multimodal Information

In the late 20th century the flood of information is not only increasing but diversifying. Information is now available in a multiplicity of media that includes texts, photos, video clips, and films. The diversity and volume of accessible on-line information from a variety of sources, including the Internet, is increasing dramatically. Not only are unstructured data and structured data available in abundance, but non-text data such as images, audio and video clips are increasingly becoming available. This increase in diversity and availability is a new phenomenon and it poses new problems that have not yet been addressed.

In particular, when information is available in diverse sources, exchanging and coordinating information between these sources is a major challenge. For example, exchanging meteorological information between a satellite photo, a relational data base encoding meteorological measures, a schematic map and a report in English is a task that requires a high degree of expertise. Somebody without this expertise would not know to what extent these sources share information. Thus, if an organisation is to maximise its use of information available in a variety of sources, it will have to meet the challenge of exchanging and coordinating meanings between the different information sources. This is the challenge of managing multimodal information.

1.1 Multi-Media: A Solution?

The technology of multi-media does of course enable a user to create a package that connects different sources of information through reference links, but it does not assist in the translation between these different representations. Rather, the creator of a multimedia package has to handcraft all the reference links, and the user will only be able to explore those pre-existing links. This is so because the links are created at a low level of representation and also because a unified theoretical interpretation of such systems is yet to be developed.

2. Theoretical Approach

The technology of multimodal information processing runs far ahead of our theoretical understanding of the field. However, a theoretically unified approach to multimodality is possible based on systemic-functional theory and has been applied to language as well as other semiotic systems (Ravelli, 1995; Matthiessen, Kobayashi and Zeng, 1995). Matthiessen et al (1995) apply the theoretical approach to the problem of multimodal weather forecasting which may be taken as a prolegomena for such research in the Defence organisation.

Following the systemic-functional model, the resources of meaning in semiotic systems including language, are organised according to three metafunctions: the interpersonal, the textual and the ideational (Matthiessen and Halliday, 1996). The interpersonal metafunction provides resources for representing the relationships between participants. In language part of that resource is the way in which an exchange between speaker and listener is possible (Halliday, 1994). In the semiotic of visual images, the interpersonal metafunction is the resource for setting up the relation between the producer and the viewer of the image (Kress and van Leeuwen, 1990). The textual metafunction provides the resource for structuring the message. In language, part of the resource consists of Theme which is the point of departure for the message. In visual imagery, the resource provides the means by which images are composed (ibid.). The ideational metafunction provides the resources for modelling our experience of the world (refer § 3.1). In language we are concerned with the world configured as process, with participants in that process and the circumstances of that process. In visual representation we are also concerned with participants, processes and circumstances, but in this case of images (refer § 4.1). In the current phase of the research, the work is focussing on the ideational resource and exploring how experience is construed in different modalities.

3. Multimodal Information in Defence

The Defence Organisation has to meet the challenge of multimodal information because like most organisations, the majority of Defence is required to integrate information that comes in a variety of forms, and covers a variety of domains.

3.1. Geospatial Dimension

One pervasive and uniting theme of the domains that interest Defence is the *geospatial dimension.* This dimension can be used to illustrate both the problem needing to be addressed and the significance of the research proposed here.

Like other facets of human experience in a given domain (such as health or meteorology), the geospatial dimension is construed by the **ideational** resources of language — the resources for modelling our experience of the world around us and inside us **as meaning**. Thus our experience of the 'flow of events' is modelled as quanta of change organised as configurations of a process, participants involved in this process (bringing it about or being affected by it), and circumstances of time, space, cause etc. associated with the process. The geospatial dimension is modelled in terms of circumstances. For example, in the following extract from a text, geospatial circumstances are in italics, temporal circumstances are underlined and processes are indicated by bold:

> Trypanosmiasis. Foci **occur** *in the Akagera Game Park (northeast)* and presumably **exist** *in the Nasho Lake vicinity (east of Kigali).* Current incidence data **are** not available, but sporadic cases of rhodesiense disease **were reported** *among foreign travellers to the Akagera* <u>during the late 1980s</u>, and sporadic cases **were reported** <u>during the early 1980s</u> *from the Akagera Game Park and the Nasho Lake vicinity.*

This text is centrally concerned with the geospatial location (e.g. *in the Akagera Game Park*) of a disease (construed as a participant: *rhodesiense disease*) and also with temporal location (e.g. *during the late 1980s*).

Alternatively, certain aspects of the meanings presented above as English text, particularly the geospatial meanings, could be presented by means of a map. The map with appropriate icons and captions could show both absolute locations (e.g. *in the Akagera Game Park*)) and relative locations (e.g. *northeast*). It is also possible to represent some other aspects to the meanings of the text; for example, the disease foci (construed as participants) could be represented as graphically foregrounded areas (with appropriate key where the areal patterns are glossed in English). However, other aspects of the meanings are less likely to be represented cartographically — aspects such as processes other than those of existing (being) in an area (e.g. be reported) and the location of processes in time. Moreover, type information is in general hard to express cartographically, whereas language is a rich resource for constructing taxonomies, such as taxonomies of diseases.

3.2. Multimodal Domains in Defence

In Defence there are many domains where information is multimodal. At a minimum Defence domains utilise the modes of map based information and text. However, in many of these domains some, if not all, of the information is classified and therefore not freely available for research purposes. The domain in which the current research is concentrated is fortunately largely unclassified as was evidenced by the example in the previous section. It is the domain of health and medical information as it relates to the Australian Defence Force. In its broadest definition the domain is concerned with any health/medical related information that may affect a country's national interests. In the present day context of global peace keeping, the domain covers not only local, but potentially world wide, health information. Thus the domain includes information that facilitates the tracking and monitoring of diseases.

4. Multimodal Research in Defence

As part of a current research project, a multimodal meaning base is being developed for the health and medical domain that not only makes it possible to identify meanings that can be

expressed both by particular linguistic and by particular cartographic conventions but also those meanings that can only be expressed by one modality or the other.

Ideally, it would be possible to cover all health and medical information relevant to this domain but given the scope of the research, a limited number of diseases have been selected with which to experiment. These diseases include 1. the Ebola virus 2. Japanese B Encephalitis 3. Dengue Fever and 4. HIV. The selected diseases form what may be called probes. The probes permit testing of multimodal resources.

The research project also involves the collection of multimodal resources to form a corpus of different types of sources. The sources include structured information, for example, databases, and unstructured information, for example, texts, maps, and other visual information, such as photographs and schematic diagrams. An example will be given of how meanings in the domain are tracked across a multimodal corpus.

4.1. Tracking Meanings across Multimodal Sources: An Example

Tracking meaning across the multimodal corpus will be illustrated by means of the Ebola Virus probe. Excerpts from the multimodal corpus are given in the following figures. Figures 1 to 5 give text extracts from various sources describing various aspects of the Ebola virus. Figure 6 is a map extracted from one of the sources showing Zaire and its capital Kinshasa. An Ebola outbreak occurred in Kikwit a city located 240 miles east of Kinshasa (refer below). Figure 7 is a photo (an electron micrograph) of the virus plus accompanying caption. Figure 8 is an excerpt from a database design for diseases.

A variety of meanings are available in the excerpts including geographic location, disease characteristics and the cause of the disease i.e. the pathogen. Let us begin by examining how geographic location is construed in the various sources, and then divert to the disease and pathogen meanings for further illustration. The construal is given in less rather than more detail, with the aim of conveying the flavour of the process, rather than an in depth analysis.

In the first extract, figure 1, the first and only participant is *Zaire* standing alone in a minor clause. In the second clause, again a minor clause, two participants occur *Capital:* and *Kinshasa,* but where one might expect a relational process linking the two participants that is omitted. While this is not the current focus of the paper, one would examine the register of the text in order to make sense of the particular choices made, for example, minor rather than major clauses.

Zaire

Capital: Kinshasa

Figure 1: Extract: Zaire - CIA World Fact Book

In the second extract, the geographic location is represented as circumstance of place *in Kikwit, Zaire* and an embedded minor relational clause *a city located 240 miles east of Kinshasa* viz. the underlined text fragments.

Title: Outbreak of Ebola viral haemorrhagic fever - <u>Zaire</u>

Abstract:

On May 6 1995, CDC was notified by health authorities and the U.S. Embassy in Zaire of an outbreak of viral haemorrhagic fever (VHF)-like illness <u>in Kikwit, Zaire</u> (1995 population: 400,000), <u>a city located 240 miles east of Kinshasa</u>. The World Health Organisation and CDC were invited by the Government of Zaire to participate in an investigation of the outbreak.

Figure 2: Extract: Outbreak of Ebola - Morbidity Mortal Weekly Report 1995 May 19; 44(19): 381-2

The next textual extract, figure 3, does construe some meanings concerning geographic location of the disease but this time the meanings are linked to the participant of a relational process *named for* .

<u>Ebola Virus Haemorrhagic Fever: General information</u>

What is Ebola virus?

The Ebola virus is a member of a family of RNA viruses known as filoviruses. When magnified several thousand times by an electron microscope, these viruses have the appearance of long filaments or threads. Ebola virus was discovered in 1976 and was named for <u>a river in Zaire, Africa, where it was first detected</u>.

Figure 3: Extract 1: Ebola Virus - CDC World Wide Web

In the next extract, figure 4, geographic location is first construed as circumstance to a relational process *in Kikwit, Zaire* and then detailed as a circumstantial relational process *located* with accompanying participants. Note how it is possible to build in other identifying information using a relational clause viz. *Kikwit is a city of 400,000.*

<u>Ebola Virus Haemorrhagic Fever: General information</u>

What do we know about the recent outbreak of Ebola virus infection?

The recent Ebola virus outbreak is centred <u>in Kikwit, Zaire. (Kikwit is a city of 400,000 located 400 kilometres east of Kinshasa, the capital of Zaire.)</u>

Figure 4: Extract 2: Ebola Virus - CDC World Wide Web

The next textual extract deals with meanings describing the disease itself and is included for contrast with the construing of geographic meanings. Note that in this extract there are human participants *Persons* and *patients* , and nonrelational processes viz. *develop* and *bleed*.

What are the symptoms of Ebola Haemorrhagic Fever?

Symptoms of Ebola haemorrhagic fever begin 4 to 16 days after infection. Persons develop fever, chills, headaches, muscle aches, and loss of appetite. As the disease progresses, vomiting, diarrhoea, abdominal pain, sore throat, and chest pain can occur. The blood fails to clot and patients may bleed from injection sites as well as into the gastrointestinal tract, skin, and internal organs.

Figure 5: Extract 3: Ebola Virus - CDC World Wide Web

In summary, for the textual sources, geographic meaning is construed as circumstance and circumstantial relational process both in minor and major clauses.

The next extract is in a different semiotic, that of visual imagery, and in the register of maps. In figure 6 the map is a schematic image showing the relative distribution of named places. Linking to the location of the outbreak of the Ebola virus is *KINSHASA*. The map represents the participants, the named places, in a "conceptual" location, that is, the more or less timeless, stable and constant visible locational essence of the places is portrayed (Kress and van Leeuwen, 1990). One may note how the status of the capital is differentiated from the other named places by use of capitals (part of the textual resource for meaning), but that convention is not given in a key to the map. It would have been possible to choose a representation of location that was "presentational" showing how the participants relate to each other in a given specific instance. In such a case an aerial photograph of Zaire may have been chosen.

Figure 6: Map of Zaire - CIA World Fact Book

Indeed it may be argued that the next image, that of the Electron Micrograph of the Ebola Virus, figure 7, is a choice of presentational process to represent meanings concerning the

pathogen of the disease. Here the image is of an actual specific instance of the virus, rather than, for example a labelled diagram which would be a representation as conceptual process. However, it might be argued that in the caption the textual resources capture both the conceptual, which is in linguistic terms a relational process *-Electron Micrograph of Ebola Zaire Virus* , and the presentational process of a particular event - *Diagnostic Specimen in cell culture at 160,000 x magnification.*

Electron Micrograph of Ebola Zaire Virus. This is the first photo ever taken, in 1976, by Dr. Frederick A. Murphy, now of the University of California - Davis, then director of the National Centre for Infectious Diseases. Diagnostic Specimen in cell culture at 160,000 x magnification.

Figure 7: Electron Micrograph of Ebola Zaire Virus-Extract from Access Excellence

The final extract also captures meanings related to the pathogen but this time in the structured form of a database design.
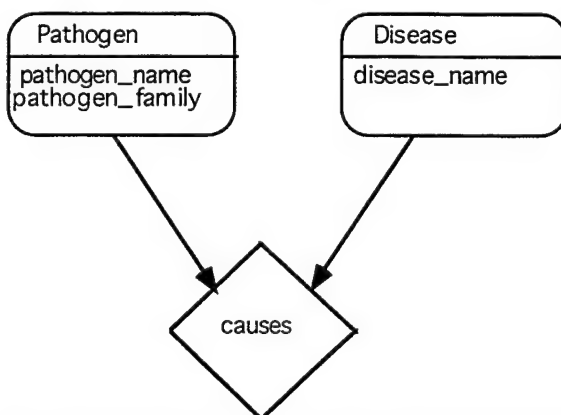

Figure 8: Extract of Design for Disease Database

In figure 8, the participants are captured in both the naming of the entities *Pathogen* and *Disease* and in the (database) relation between the entities *causes*. The representation is not only textual but also graphic (in the terms of Kress and van Leeuwen, it is a visual image). As a visual image, the objects of the image - the shapes and arrows and their relative placement, also contribute to the meanings. The ideational meanings in the image are represented conceptually i.e. they attempt to capture the stable and constant qualities of the domain.

5. Summary

Organisations today are faced with the challenge of managing multimodal information. Defence is no exception and, indeed, faces the particular challenge of including information that is construed along the geospatial dimension. The technology of multimedia provides a means of connecting different sources of information using reference links, but it does not assist in the translation between these different representations. An exemplar domain of health and medical information is described that has been selected for multimodal research in Defence. A discursive analysis using the systemic-functional approach is presented using extracts from different types of sources, text, maps, images and database. Using the meaning probe of the Ebola virus, meanings, such as geographic location, disease characteristics and disease pathogen, are tracked, exploring how the ideational meanings are construed in the different types of sources.

References

Halliday M. A. K. 1994. *An Introduction to Functional Grammar*. London: Edward Arnold.

Kress G. and van Leeuwen T. 1990. *Reading Images*. Deakin: Deakin University Press.

Matthiessen C. M. I. M. and Halliday M. A. K., 1996. *Construing Experience*. In Press.

Matthiessen C. M. I. M., Kobayashi I., and Zeng L. 1995. Generating Multimodal Presentations: Resources and Processes. *Proceedings of Workshop , Artificial Intelligence in Defence, Eighth Australian Joint Conference on Artificial Intelligence (AI'95)*, Canberra, November, 1995.

Ravelli L. J. 1995. Intersemiosis: the Constraints and Potential of Verbal-Visual Interaction. *Paper Presented at the 22nd International Systemic-Functional Conference*, Beijing, July 18 - 22, 1995.

The Modelling of Multimodal Resources and Processes for Defence

Christian MATTHIESSEN*, Ichiro KOBAYASHI ** ZENG Licheng* and Marilyn Cross***

* Macquarie University

** Hôsei University

***DSTO

1. The Challenge of the Modern Information Environment

The modern military commander is flooded with information in a multiplicity of media that includes texts, photos, video clips, films. Multimedia has become both a buzz word and a buzz product. It is here for a reason: Defence is being convinced to consume multimedia information and multimedia products, living at the cutting edge of technological developments. At the same time, it seems that technology is running well ahead of theoretical understanding. Or rather: if the focus of the current technological innovations is information, it would seem that the breakthroughs have been the at lowest levels of information systems — the levels of expression in different media, where the concern is how to digitise these diverse media. To be somewhat provocative, we are only producing and consuming somewhat more sophisticated versions of what was produced half a millennium ago, at the dawn of the modern scientific period. The material medium has change from that of the printed page. But what about the higher levels — the realm of meaning (or "knowledge", to look at it from a cognitivist point of view)? These have arguably not begun to be targeted yet.

If we consider the average multi-media product, we will find features such as the following:

- Traditional cross-referencing has been "implemented" so that instead of having to turn pages to find the cross-referenced location, we can just click on the cross-referenced items.

- Non-linguistic presentations are glossed linguistically so that they can be entered into lexical taxonomies and searched.

For example, a text entry in a multimedia encyclopaedia may contain a link to an image (a photo, drawing, painting or map), a video clip or a sound; and that image etc. is indexed linguistically and has a location in a linguistic inventory or taxonomy of images. However, beyond this classification of images, there is no representation of what they mean. Therefore the encyclopaedia has no model of how the different images complement one another in giving meanings to the user: it has no model of how the presentational labour should be divided between text and image. In other words, the system of meaning that lies behind the construction of a multimedia presentation has not really been modelled. One of the consequences of this seems to be that the way in which images are used is basically only an electronic implementation of what we find in a traditional encyclopaedia.

The absence of a meaning base supporting a multimedia system such as a multimedia encyclopaedia becomes very clear if we consider what it would take to generate multimedia entries in an encyclopaedia automatically, This is the kind of task that has begun to be addressed in work on **multimodal** generation systems (e.g. Feiner and McKeown, 1989; Wahlster et al, 1992). Here we can begin to explore the meaning-making potential of a multimodal system.

In this paper, we will briefly sketch a unified theoretical interpretation of multimodal systems (Section 2) and in the remainder we will present a more detailed design of a multimodal system using weather as a component of environmental reports as our example. In Section 3, we describe a particular weather report and discuss the division of labour between text and image. In Section 4, we model the multimodal meaning base of weather reporting and comment on aspects of content planning. Having suggested an approach to multimodality, we relate it to relevant work in the area (Section 5).

2. Multimodal Displays

We can break the problem of multimodality into two subproblems:

[i] How do we construe the individual systems included in a multimodal system?

[ii] How do we co-ordinate and integrate these systems while retaining their individual integrity?

The solution to the second problem depends on how we address the first problem, so we will consider that problem first.

2.1 Construing Non-linguistic Semiotic Systems

The systems involved in a multimodal system are all systems of a particular kind: they are **semiotic systems**. That is, they are systems of **meaning**. The special character of such systems can best be seen by locating them in a typology of systems (see Halliday, 1995; Halliday & Matthiessen, forthcoming). In this typology, semiotic systems are systems of the highest order:

[i] First order systems are **physical systems**: such systems are subject to physical laws.

[ii] Second-order systems are **biological systems**; such systems are physical systems with the addition of life.

[iii] Third-order systems are **biological systems**; such systems are also biological (and so physical), but with the addition of value or role in a network or relations.

[iv] Fourth-order systems are **semiotic systems:** such systems are also social (and so biological and physical), with the addition of meaning.

Figure 2-1 represents the typology as an ordering from physical to semiotic. All semiotic systems embody meaning; this means that they are **stratified:** minimally, they are organised into two strata (levels) of abstraction — content [meaning] and expression. The standard example is that of traffic lights: this simple semiotic system consists of a small set of content/ expression pairs such as stop/ go and drive/ green.
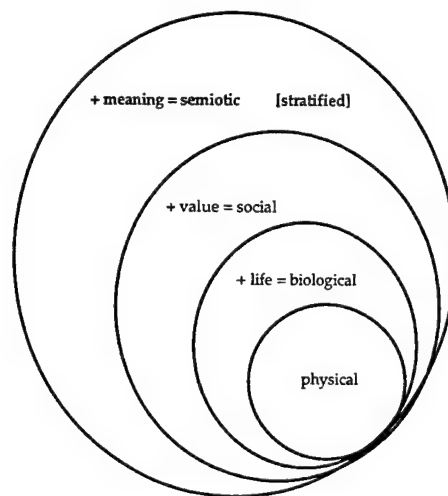

Fig. 2-1: Ordering of kinds of system

A multimodal system is thus a system of systems of the same order: it is a system of semiotic systems; and a more revealing term for such systems would arguably be multi-semiotic systems. Consequently, the systems in a multimodal system will at least share one property as systems of meaning, they will be stratified into content / expression. Identifying such general properties is quite relevant to the task at hand since[1]

---

[1] We confine ourselves here to (human) social semiotic systems since it is such systems that are relevant in multimodal models. We will thus not consider semiotic interpretations of other orders of systems, in particular biological systems and physical systems, as in

- common properties can be modelled in the same way for all the semiotic systems involved (which has implications for the treatment of multimodality);

- common properties might be able to be modelled on the basis of one representative semiotic system that has already been investigated in considerable detail — viz. language.

The first assumption would seem to be quite uncontroversial, but the second assumption is perhaps more controversial. However, there are sound reasons for making the assumption. Language is the primary and prototypical human semiotic.

2.2 Construing multimodality

The system network (Halliday, 1966; Halliday, 1976) makes it possible to abstract away from the constraints inherent in the expression system of different semiotic systems. Consequently, even though we will need different realisation ("rendering") statements for different expression systems, we can model the meaning potentials of different semiotic systems in terms of the same type of systemic organisation. This makes it considerably easier to compare and contrast the emanating potentials of different semiotic systems.

2.2.1 The meaning potential of a multimodal system

But how similar would the meaning potentials of different semiotic systems be? This is an empirical issue If we are to develop a description of the systemic potential of maps, they would of course have to be different in various respects to the descriptive categories that have been identified in the account of the meaning potential of English (see Halliday, 1985; Matthiessen, 1995).

In our approach, the different semiotic systems are integrated into a single, coherent meaning potential. This integration brings out the commonality of the different semiotic systems but at the same time, the individual systems are integrated in such a way that the integrity of each system is preserved. The basic principle is quite straightforward. The meaning potential is represented by a system network, where commonalities across semiotic systems are simply represented as shared parts of the network whereas meanings that are not shared are represented within **partitions of the network**. This will be illustrated in detail in Section 4 below. Partitioning the system network introduces conditionalisation on systems (or system parts or realisation statements) — or rather, meta-conditionalisation, since the conditionalisation is external to the logic of the system network itself. Conditionalisation can be linked to contextual features.

Modelling the semantic domains of the various semiotic systems as an integrated meaning potential makes it easier to generate multimodal presentations:

- a text planner can reason with the integrated meaning potential uniformly. When a request to present some information is sent to the planner, the planner has at its disposal to present the information in text, map or text-map combination.

- a text planner is aware of what can be done or what cannot be done in each semiotic system, and hence is able to distribute information to be presented to text and weather maps.

- a text planner is able to manage the coherence (conjunctive relations or referential relations) between text and resources such as maps.

The approach to multimodal modelling just sketched makes it possible to investigate the degree to which different semiotic systems have congruent meaning potentials. We will illustrate how this can be done in Section 3 below. Here we will only note that it seems reasonable to us to take as one's base the accounts of the meaning potential of the language or languages to be included in a multimodal system. Language is undoubtedly the most powerful and the most complex of all human semiotic systems. It has evolved as a resource for construing *all* human experience of the world around us and inside us as meaning and for enacting *all* human social roles and relations as meaning. No other human semiotic can carry this functional load; other semiotic systems are much more restricted in their range of uses than language is. As Sugeno (1993) has pointed out, language serves to integrate or fuse information from a variety of sources — both other social semiotic systems and our individual perceptual systems.

---

semiotic interpretations of neural organization and explorations of meaning exchange in physical systems.

2.2.2 Context in a multimodal system

We have suggested how the meaning potentials of different semiotic systems can be integrated. But how would the labour between them be divided in a multimodal presentation? To deal with this issue, we have to model the **context** in which the different semiotic systems serve. It is this context that will determine the division of labour between language and other semiotic systems. The division of labour is **variable:** in certain types of context, language dominates and other semiotic resources are brought in only occasionally to support the textual presentation (e.g. those in which personal letters are produced and read), whereas in others other semiotic systems are favoured and language is brought in only to facilitate by glosses, legends, keys and the like (e.g. those contexts where cartographic reference materials are produced and consulted).

Context involves three major variables — field, tenor and mode (e.g. Halliday, McIntosh & Strevens, 1964; Halliday, 1978; Halliday & Hasan, 1985):[2]

field — field of discourse: this includes the socially recognised activities (e.g. instructing in skills, coordinating collaborative work, entertaining, regulating behaviour) and the subject matter created by the semiotic systems in the course of realising these social activities (e.g. culinary, financial, meteorological) along the cline from commonsense (folk) to uncommonsense (scientific);

tenor — tenor of relationship, the (network) of social roles and relationships, specified in terms of: (i) power (by reference to authority): equal [peer]/ unequal; (ii) power (by reference to expertise): expert to expert/ expert to novice etc.; (iii) institutional role: parent to child; supervisor to staff; teacher to pupil; mate to mate; etc..; (iv) familiarity [contact]: intimates ... strangers; (v) affect [emotional charge]: neutral/ charged (positive [consensus]/ negative [conflict]); and finally (vi) roles constituted by [denotative] semiotic system, such as commander to complier; and

mode — the role played by the [denotative] semiotic systems in the context: (i) the medium (fixed for most semiotic systems, but variable for language: spoken/ written [or gestured in sign languages]) and channel (aural/ visual; face to face/ telephonic; etc.); (ii) role played relative to field (cline from constitutive to ancillary); (iii) division of labour among [denotative] semiotic systems (mono-modal/ multi-modal: e.g. text constitutive and image ancillary/ image constitutive and text ancillary); and (iv) rhetorical mode (e.g. didactic, informative, narrative, persuasive, regulatory, exploratory).

The division of labour between language and other [denotative] semiotic systems is an aspect of the mode in relation to other mode factors — that is, it is an aspect of the role they play in the context. For example, if the rhetorical mode is narrative and the medium is written, then there is a range in the division of labour between image as entirely constitutive, as in picture books for small children, to language as entirely constitutive as in adult fiction. Within this range, we find different combinations, e.g. image and language as co-constitutive, running in parallel as (partial) restatements of the same narrative (narratives for children written to be read aloud by a care-giver) and language as constitutive with image as ancillary illustration. And we can find variants of the "same" narrative distributed within this semiotic space (as with classics, folk tales and myths).

However, these different mode settings correlate with different values within field and tenor. While language ranges across all domains of subject matter within field, other semiotic systems are more restricted. Obvious examples include the use of particular image systems such as those maps, to be discussed below. Perhaps less obvious is the effect of the move from commonsense (folk) models to uncommonsense (scientific) models within field: along this cline, language is always the primary resource, but as we move away from commonsense models, diagrams are brought in to represent those aspects of experience that are construed as abstract space in our commonsense models (e.g. graphs of scales running from 'high' to 'low', block diagrams and network diagrams of memory as a space). Such field differences also correlate with tenor differences, in particular, differences in expertise.

3. Multimodal Environmental Assessments: a case study

We have now outlined certain critical features of a model of multimodal systems based on a theory of semiotic systems in context. We will elaborate and apply this model to the domain of environmental assessments, and more specifically weather forecasting, illustrating how a multimodal presentation generator draws on integrated **meaning-making** resources to produce coherent multimodal text. Weather reporting has been selected as it forms a component of environmental situation assessment that is necessary for command and control but with the advantage that examples can be drawn from unclassified sources. In the example given

---

[2] We follow these works but try to generalize field, tenor and mode to include not only language but also other semiotic systems.

the weather reports have been taken from newspapers, a totally unclassified source, but the extrapolation to military classified sources may be made. Section 4 shows how the semantic resources of environmental texts and maps can be integrated into a single coherent meaning base.

3.1 The register of weather forecasting

Weather reports include contributions from different semiotic systems — text, images: maps, and tables. Modelling the shared meaning base and the generation process in such a domain is manageable. Moreover, automating the generation of presentations within this kind of multimodal register is clearly desirable: there is a steady flow of incoming information and the task of producing regular reports is quite repetitive.

As a register, weather forecasting includes a number of subregisters differing in *mode* (written, in reports, accompanied by images; spoken, in radio broadcasts, on closed-circuit TV, accompanied by images, sometimes with some animation), in *field* (nature and technicality of meteorological information) and in *tenor* (general military/ or special interest groups). All these subregisters are not necessarily multimodal: in radio broadcasts, the spoken text has to constitute the entire forecast; within the register of weather forecasting, the division of labour between different semiotic systems is thus variable. At the same time, the register of weather forecasting is related to other registers involving reports on and forecasts of measurements — e.g. stock market reports, exchange rate forecasts and reports on disease patterns.

3.2 Analysis of a multimodal weather report

We will focus on a standard weather report (refer figure 3.0) and sketch a brief systemic-functional description of it. Like weather reports in general, this report consists of contributions from three semiotic systems: weather texts which describe weather conditions; weather maps which project meteorological phenomena onto geographic space (at some interval in time); and weather tables which intersect meteorological measures (in particular, temperatures) with geographic locations and meteorological probabilities (of rain occurring) with time periods.

3.3 Contextual description

We will consider the context of the register of weather forecasting. We introduced the major contextual variables in Section 2.2, field, tenor and mode, and can now use them to characterise the situation type in which weather reports are produced:

- **Field**: disseminating present and future weather conditions by daily newspaper, specifying (1) locality, from district to world with emphasis on regions (2) causes of meteorological events (3) potential cautions for special interest groups, eg. farmers, fishermen and aviation etc.

- **Tenor**: expert standpoint — impersonal, with acknowledgment of uncertainty; audience — general public plus some special interest groups.

- **Mode**: written, monologue, accompanied cartographic presentations (including satellite photos overlaid by maps).
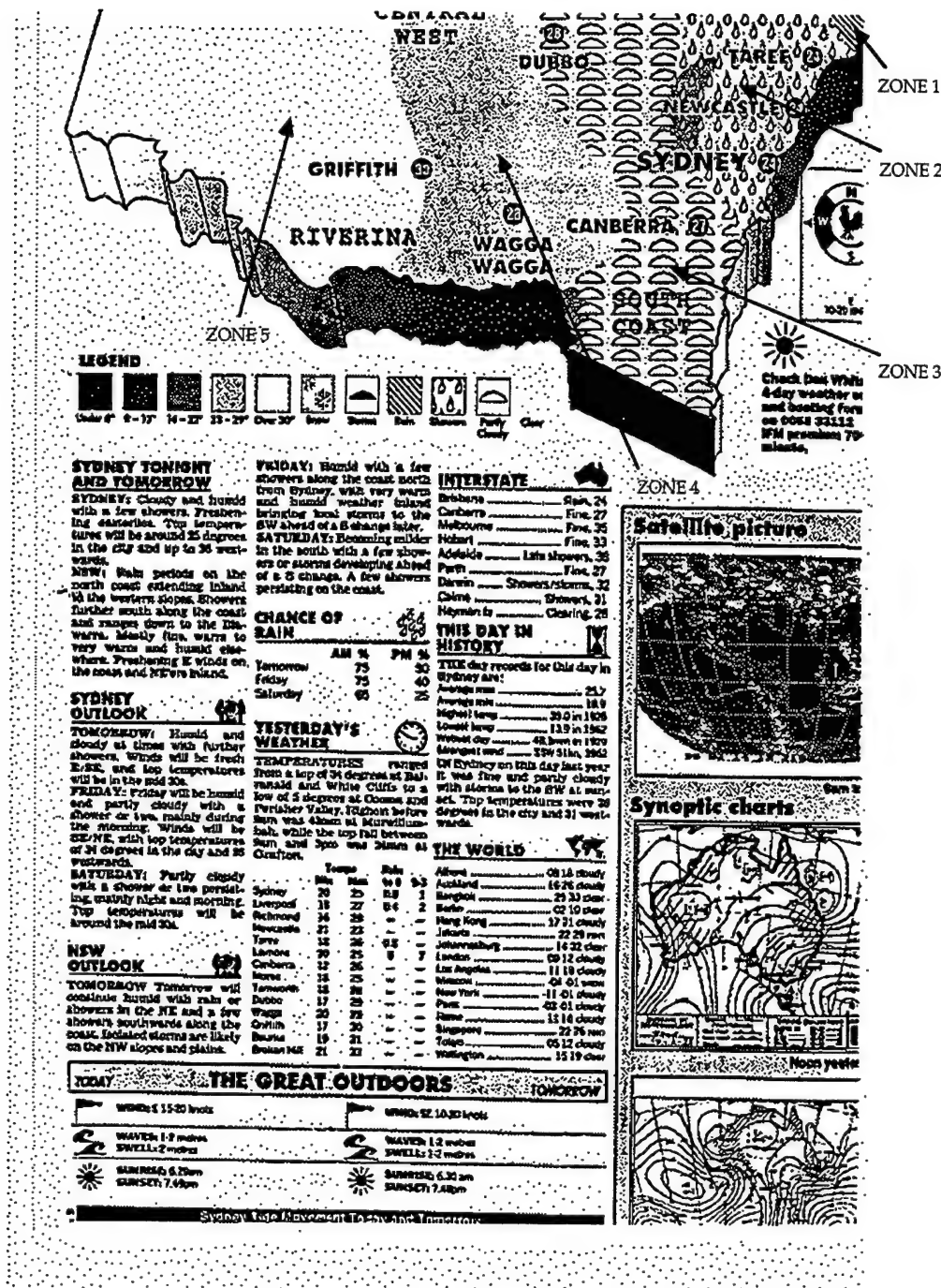
Fig. 3-0: An example of a weather report

3.4 Semantic description

The situation type sketched above embraces all the semiotic systems involved in the register of weather forecasting. The field is realised by selections of ideational meanings within these semiotic systems.

The ideational resources are used to construe meteorological phenomena as meaning. Here the systems of language and of maps overlap considerably. Both construe a more or less same set of meteorological phenomena. The example in Figure 4.1 illustrates this overlap in the information expressed by text and weather maps. It also illustrates the division of labour between the semiotic systems. The principle is that while weather maps are generally used to present a snapshot of weather conditions of all regions at a fixed

temporal location, language is exploited to highlight weather conditions at certain regions and to indicate prediction and change of weather conditions. Let us consider (i) texts, (ii) maps and (iii) tables in more detail.

(i) Weather texts

• For general public:
(Sydney tonight and tomorrow, Sydney outlook, NSW outlook)
    The text predicts the weather conditions of the day's evening and of the following day, The text describes future weather conditions, changes in the weather, and meteorological causes.

• For special interest groups:
(Weather reports for the purpose of general and special)
    The text also includes information about weather conditions needed for special-interest activities such as surfing, skiing, and fishing.

(ii) Weather maps

    Generally speaking, maps project synoptic weather conditions onto geographical space during some temporal period.

• *Simple weather map (New South Wales)*
    This weather map is a schematic image. It shows the weather conditions of local cities in NSW, indicating the temperature of the cities.

• *Atmospheric pressure distribution maps*
    This map is a schematic image. It shows the distribution of atmospheric pressures on Australia. And usually some distribution maps are presented simultaneously to indicate the history of the motion of atmospheric pressures.

• *Satellite map*
    This map is a photographic image enhanced by contour lines. It shows an overview of the weather conditions that can be encoded photographically from a satellite: cloud patterns and coverage. The amount of information that can be read or inferred from it depends on the reader's expertise.

(iii) Weather tables

    Generally speaking, tables represent sets of relations; they relate meteorological values to geographic locations (temperatures) and temporal locations (pluvial probabilities).

    Table 4.1 shows how different meanings are distributed **across the semiotic systems.**

Table 4.1: Functional characteristics of each modality

Type of meaning	Semiotic system		
	Text (Weather texts)	Image (Weather maps)	Table (Weather tables)
temperature in region	√	√	√
temperature in city	√	√	√
weather in area	√	√	√
weather at time	√		
wind in area	√	√	√
prediction of weather	√		√
cause of weather	√		

We can see that certain types of meaning are restricted to the textual presentation. For example, text can express the cause of weather in a weather report, however, other modalities cannot do that. The text provides the most comprehensive presentation of meteorological meanings and framework in terms of which the contributions from maps and tables can be interpreted. In that respect, language is the dominant semiotic, constituting the whole field.

4. Integrating Multimodal Environmental Resources

In the previous section, we identified the salient contextual and [denotative] semiotic features of weather forecasting. We can now turn to the central task of modelling the (ideational) meaning base needed to support the generation of multimodal weather reports. We will focus on the meaning potentials of language and maps since the system of tables is very simple. To model the multimodal meaning, we have to be able to compare the semantic categories of English and maps along the systemic lines discussed in Section 2.2 above. We will proceed in a number of steps.

4.1 Step 1: map out the resources for domain sub-language

The first step is to map out the meaning potential embodied in language of weather forecasting (i.e. the linguistic part of the multimodal register), using system networks to classify the semantic categories construed in the weather sub-language. With system networks, we obtain a panorama of partitions of the meaning space accommodated in the register.

Halliday and Matthiessen (1994) partitioned the ideational meaning potential of the register into three major categories: **sequence, figure** and **element. Sequence** covers the semantic areas such as weather events sequence, various temporal or spatial relations concerning weather conditions and regions affected by particular weathers. **Figure** is "a basic fragment of experience that takes the form of one quantum of change", Here it concerns with weather status (eg. *sunny skies will dominate most of regions.*) or weather change (eg. *isolated showers developing on the South Coast later*). **Element** are constituent parts of figures — the participants, processes and circumstances involved in figures. These semantic types will be classified more delicately below and will be illustrated with some examples.

Figure 4-1 shows a fragment of the system network of **sequence**. Extending means adding new weather information to previous information. The extending information is 'new' in that it indicates a change in either time, place or weather conditions. Alternatively, a sequence may construe a relation of elaboration: some meteorological meaning is further elaborated by a summary or a commentary.
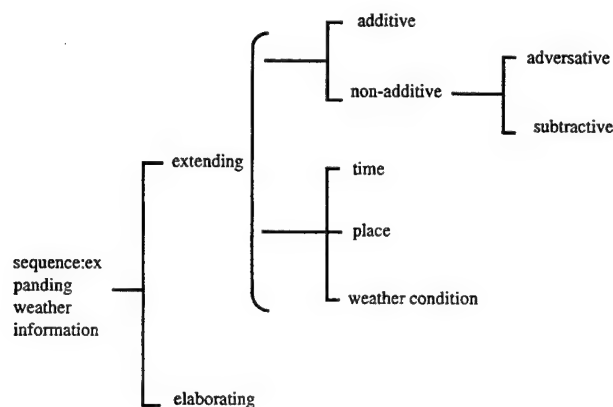

Fig. 4-1: The **sequence** system of the weather sub-language

Examples of sequence are:

{ extending additive weather-condition}
The chance of showers will end by Sunday night, and winds will shift to north.

{ extending non-additive adversative time }
Morning skies will be partly cloudy today, becoming partly sunny by afternoon.

{ elaboration commenting}
high temperatures will be 70s to low 80s, warmest in NSW.

The system of **figures** is shown in Figure 4-2. **Figure**s are primarily concerned with a change or a status of weather condition. They configure various weather participants in a configuration (eg. a weather condition in a region). In addition, we can express the causality, the phase and our prediction of the status or change of weather conditions.
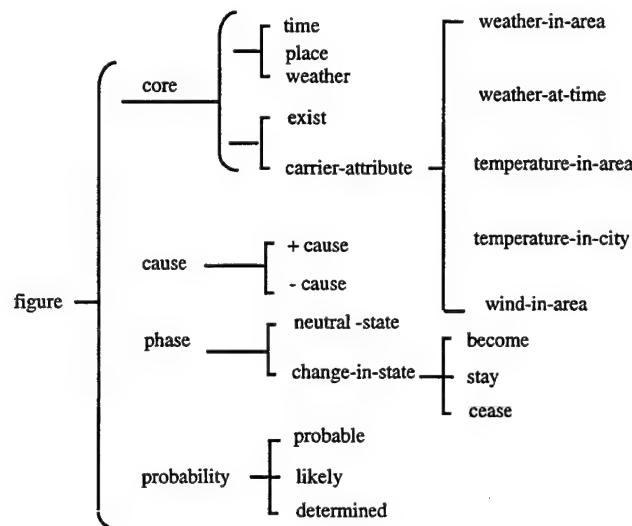

Fig. 4-2: The **figure** system of the weather sub-language

Examples of figures follow.

{core time carrier-attribute -cause neutral-state determined }
tomorrow will be mostly Fine and an expected maximum of 26.

{ core weather carrier-attribute weather-in-area   +cause neutral-state probable }
a warm front may bring scattered showers or thunderstorms to the northern Tennessee Valley.

{ core place carrier-attribute temperature-in-area   -cause change-in-state stay determined }
northern Florida will remain hot.

Elements are the constituent parts of figures. An element is a *process*, *participant* and *circumstance*. The type **process** may be further partitioned into processes of causing change, of ascribing and of phasing. **Participants** construe meteorological objects, temporal intervals and geographical regions. The type **circumstance** includes temporal-spatial locations and extends. Figure 4-3 shows a fragment of the system of **elements** (as above, for this register only).
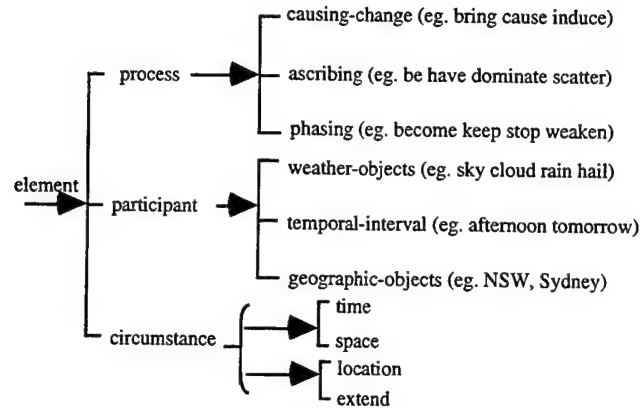
Fig. 4-3: The **element** System of the weather sub-language.

4.2 Step 2: map out the resources for domain

In step 2, we shall map out the meaning potential of the other major semiotic system. In the cartographic semiotic. we restrict ourselves to maps of the type given in the example. There are two considerations involved in extending the model of the meaning base from the linguistic system to the map system:

- as discussed in previous sections, since language provides an overarching semantic framework for other semiotic systems, it follows then that the weather sub-language outlines some semantic space into which the weather map semiotic system fits[3]. Thus we expect to see some semantic categories found in language construed in different ways in the map system.

- the weather map system, as a semiotic system, will share a similar organisation to the weather language. That is, the weather map system is organised paradigmatically as a network of types and subtypes of the potential meanings expressible through weather maps. Each type of meaning in weather maps is realised by some graphic operations, in the same way as linguistic types are realised by realisation statements. The structure of weather maps are functional in the sense that it realises meteorological meanings.

The meaning potential of weather maps is diagrammed in Figure 4-4. We have pointed out that there is an isomorphism between the meaning potentials of the linguistic weather register and the weather maps, but we must map out the semantic types of weather maps 'faithfully' according to what configurations are really construed in the maps. For instance, while in language we say *"it will rain in Sydney"*, in weather maps, the meteorological event 'rain-in-Sydney' is not construed as a meteorological figure as it is in language; rather it is construed as an attribute of the geographic object 'Sydney', meaning something like "Sydney has rain".

---

[3]This is shown by the fact that we can interpret the contents in a weather map in the weather sub-language.
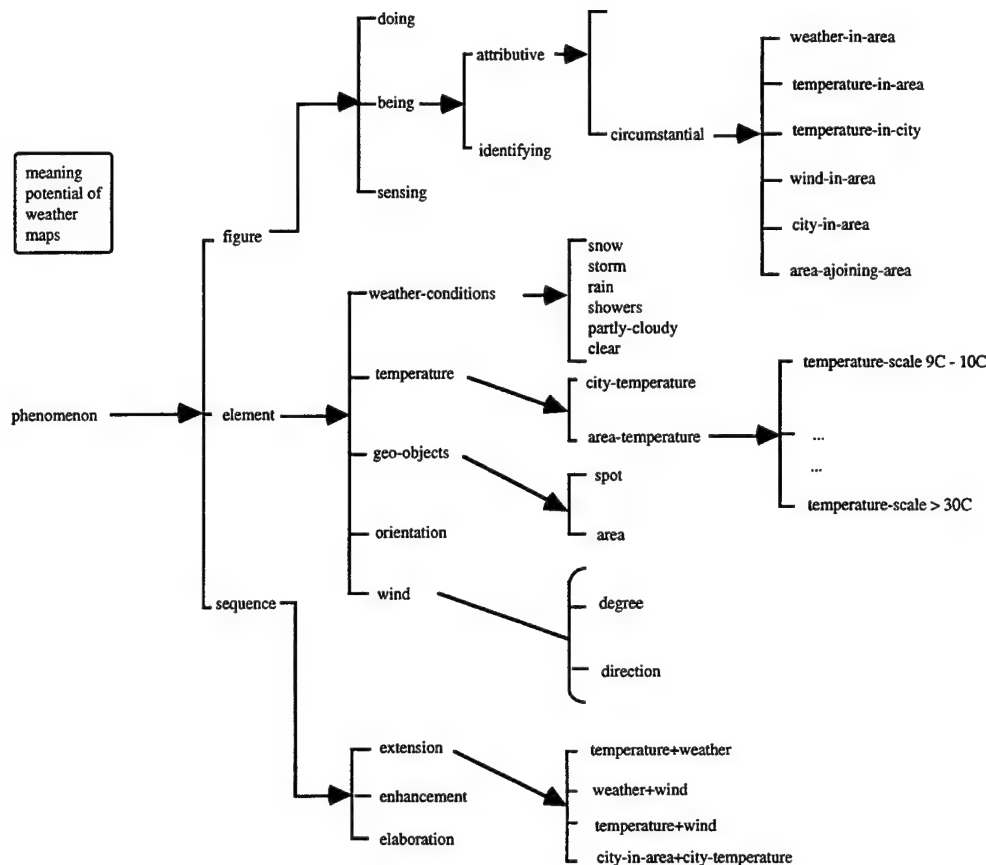
Fig. 4-4: An approximation of the meaning potential of the weather map system on the basis of the meaning potential of the linguistic part of the register

4.3 Step 3: integrate the resources of the weather-sublanguage and the weather map

Having sketched the linguistic and cartographic meaning potentials separately, we can now compare them and integrate them to form one unified multimodal meaning potential (see Section 2.2 above). In this integrated meaning potential, the functionally-equivalent semantic categories will be foregrounded and shared, but while integrity of the meaning potential of each system will still be preserved. That is, the partitioning of the meaning space of each semiotic system, i.e. the integrity of semiotic system, must be preserved in the combined meaning space.

The process of integration often involves the following two operations applied recursively across a whole semiotic system.

[1] select from the two semiotic systems semantic categories that correspond to a similar meaning as a point for merging;

[2] compare pairwise the partitions of the semantic category in each semiotic system. For each partition in one of the systems, test if it corresponds to any partition in the other system in its meaning coverage. If it does, then the two matched partitions can be shared. If not, then it is preserved in the original system. Map out the semantic gaps between non-matched partitions and provide justification for the gaps.

For example, [1] the semantic feature 'extending' in the linguistic and cartographic meaning potentials is selected because it is used in both systems to add a new piece of weather information to a previous one. [2]

We find that the linguistic system seems to observe two principles when adding new weather information. [different weather features occurring at the same temporal or spatial location] eg. *tonight will be cloudy with a chance of shower*. [same weather features occurring at different temporal or spatial locations] eg. *morning skies will be partly cloudy, becoming partly sunny afternoon*. Therefore 'extending' is partitioned into three variants: **time, place** and **weather-feature**. However since the weather map is not capable of expressing change in time or places, it is only capable of expressing the sequence of weather information equivalent to the weather-feature type in language, i.e. something like *tonight will be cloudy with a chance of shower*.

The integrated meaning potential for the feature 'extending' is shown in Figure 4-5, with the shaded area as the shared meaning area and the non-shaded area reserved for the linguistic meaning potential. Figure 4-6 represents the integrated meaning space concerning the figure feature.
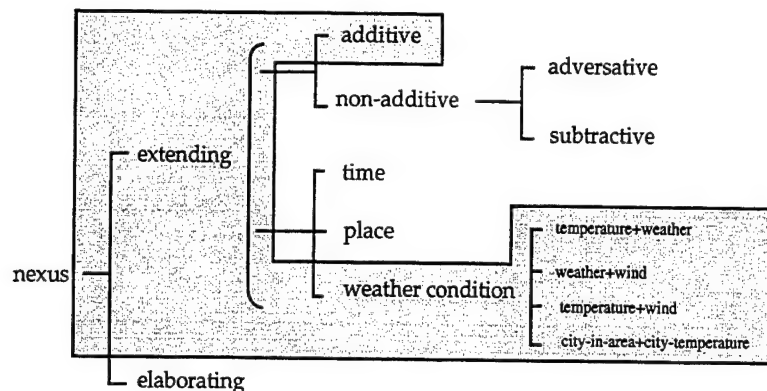

Fig. 4-5: The integrated meaning space concerning the feature 'extending '

Notice that since the expressive power of the cartographic meaning potential is quite limited, the linguistic meaning potential covers all semantic areas of the map system. The semantic gaps exhibited in the integrated system are thus one-directional; there is no semantic discrepancy from the view of the linguistic meaning potential. However, our approach allows for bi-directional discrepancies between two semiotic systems.[4]
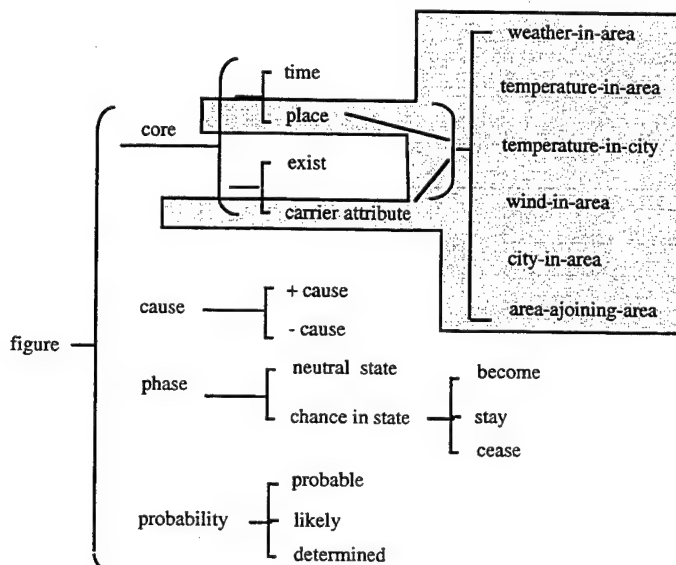

Fig. 4-6: The integrated meaning space concerning the feature 'figure'

---

[4]We have developed a multilingual text planner capable of handling the bi-directional semantic gaps. It can even compensate the gaps by reinterpreting the gaps in other terms.

It is hard to imagine that we can automate the integration of different resources, because deciding on commonality or difference among semantic categories is based on semantic judgements. We may even need to reorganise the meaning potentials to be merged to some degree in order to reveal their commonality.

In the next section we comment on content planning in multimodal generation as it is key to engineering the integration of meaning potential.

4.4   Notes on generation: content planning

Content planning is a well-known notion in text planning. In multimodal text generation, its functionality should extend to organising the contents of non-linguistic semiotic expressions as well as establishing coherence between language text and other semiotic discourse. In the case of producing multimodal weather presentations, it fulfils the following functions: (1) planning content and discourse structures of text, (2) planning content and structures of weather maps, (3) establishing coherence between text and weather maps. The process of content planning does not have to be broken down into three corresponding steps but its product, ie. the generated text-picture plan, should provide information for the three functions.

Many factors can affect content determination. The systemic theory about context provides a comprehensive account about the situation that engenders a text. For example, the field specification provides information for the planner to delimit contents of a text. The tenor information articulates the type of users and the purpose of using the generation system. The mode information specifies the rhetoric type of the text (e.g. expository versus. argumentative) as well as the media of text (eg. text, picture or combination). All these factors contribute to the choice of what to say and how to say. In our experimental system, we are not going to elaborate on the issue of input. Instead, we simply stipulate that the input to the content planner consists of only two parts: a semantic network representing a meteorological situation, and a network specifying the constraints on the content. Figure 4-7 provides an example of the constraints on the content. It can be read as follows: generating a multimodal weather presentation which on the one hand lays out the weather in NSW, and focuses on Sydney's weather and NSW's rain on the other hand. Presumably Sydney's weather situation is what readers are most concerned with and NSW's rain is a salient feature of the overall meteorological situation.
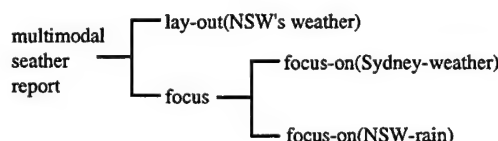

Fig. 4-7: Top-level constraints on contents

The content planner expands the content network by instantiating discourse relations that are matched by the meteorological situation encoded in semantic network and whose instantiation will satisfy the constraints maintained in the network. Therefore it plans both opportunistically and hierarchically. Given the top-level contents constraints, the planner reasons that Laying-out information is typically realised by maps, whereas focusing-on information is typically achieved by text, it thus postulates the constraints that lay-out(NSW's weather) be realised in a weather map, whereas focus-on(Sydney's weather) and focus-on(NSW's rain) should constitute a text.

As the content planner plans on, *Zone1, Zone2 ... Zone5* are selected to instantiate the relation spatial-extension (cf. logical-semantic relations in Halliday 1994 and Matthiessen, 1995), because they constitute adjacent spatial relations and can be articulated in a map, recall that the spatial relations are obtained when various data is integrated into the language-based semantics. Sydney's weather features including *humidity, temperature, weather condition* are chosen to expand the node focus-on(Sydney-weather) as they instantiate the domain of the concept *weather*. After a chain of expansion, we get the content network represented in Figure 4-8.
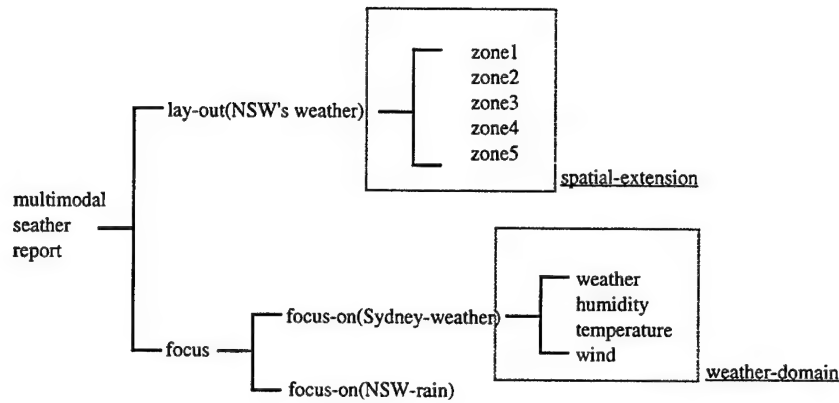
Fig. 4-8: The expanded content network after the spatial-extension and weather-domain relations are

instantiated.

Up to now, the content planner has been expanding the content network in individual map and text respectively. No coherence between map and text has been discovered yet. When *zone2* and *Sydney-weather* are incorporated into the content network, the content planner discovers an coherence between the two content nodes; *Sydney-weather* is an exemplification of *zone2*'s weather by virtue that *Sydney* is part of *zone2*, that is, an spatial-elaboration relation can be instantiated on the two content nodes. Hence a coherence relationship is established between the text and the map being developed by the content planner. The established coherence can be further expanded to include more contents, and to allow information sharing, that is, a same piece of information such as *Sydney's temperature is 23°C*, can be incorporated into both the text plan and the map plan.
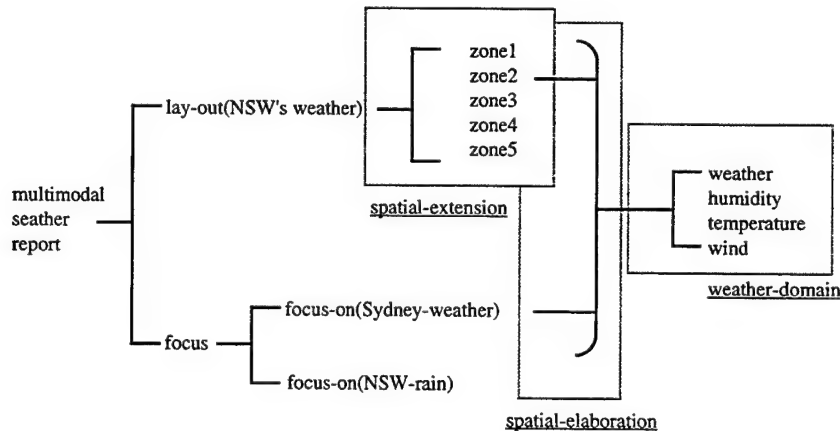

Fig. 4-9: Content network showing coherent relation between map and text

The output of content planning is the content network similar to Figure 4-9. It will be used and potentially revised by the text planning component. Due to space constraints, we can only make some brief observation on content planning here.

- the content network fulfils the three functions we require of the content planner. Its partitioning of contents for text and weather maps, as well as the established coherence between the two, appeal to our intuition about the structures of text and maps shown in Figure 4-1.

- the integrated meaning space plays an important role in content planning. Firstly, it enables a planner to plan contents for both weather map and text in a single process. Both linguistic resources and cartographic resources are concurrently probed by the planner as resources for constructing meteorological meaning. Secondly, the integrated meaning is essential for establishing coherent ties between text and maps. For instance, in determining the spatial-elaboration relation, the content planner can reason that the topological spatial-IN relation is a sub-type of the more general spatial-

<u>elaboration</u> semantic relation, thus the discourse-based semantic relation, which is normally applied to text, can be instantiated for cartographic relations.

- the relations instantiated to expand a content network include both schema (1985) and RST-like relations (Hovy 1989; Matthiessen, in press). From the standpoint of systemic linguistics, the former is more register-specific and the latter is less register-specific.

- the system network formalism is used to represent plans instead of the action network formalism conventionally used in AI planning, such as NOAH and KAMP (Sacerdoti 1977, Appelt 1985). Each 'snapshot' of the content network in the course of planning represents the potentials that constrain the future growth of text. With each plan step, some potentials are closed down and no longer pursued, while other potentials are realised and further expanded, introducing more potentials for growth. The system network formalism is preferred to the action network because it encodes both the history of expansion and the result of expansion, thus representing text both as a process and as a product (see Matthiessen 1993 for a detailed account of logogenesis).

5. A Brief Survey of Approaches to Multimodal Information Processing

Generally speaking, multimodal information processing technology runs far ahead of our theoretical understanding of the field. It is basically still a technology-oriented endeavour, combining technologies from various disciplines. In this section, we will discuss relevant techniques under three headings: multimodal text generation, visual information representation and integration of multimodal information.

5.1 Multimodal text generation

There has been a growing consensus within the natural language generation (NLG) community that in order for text generation systems to be of practical use, they must be extended to process modalities other than language. The growing interest was reflected in the multimodal document workshop held in conjunction with the international conference on text generation in Italy in 1992. We will discuss some specific research below.

Arens et al. (1988) present a multimedia interface system that automatically creates displays of naval information using a combination of maps, icons, NL-text and tables. They argue that creating multimodal text is not simply a matter of assigning various types of information to appropriate modalities. Rather, a planning mechanism is required to overlook the coherence between different multimodalities. Kerpedjiev (1992) describes a multimodal weather presentation system that can present a meteorological situation in a variety of semiotic systems including NL text, speech, graphics, tables and deictic expressions. His system exhibits two typical features foregrounded in a multimodal system: accepting multimodal information as input and tailoring modality and contents to specific user groups. Perhaps the most substantially implemented multimodal text generator to date is WIP (Wahlster et al, 1992), developed by a project that has run for several years under the direction of Wahlster. This system generates multimodal explanations and also multimodal instructions in assembling and maintaining physical devises such as an ESPRESSO machine.

A feature shared by these systems is that techniques developed for NL text generation are applied to the generation of multimodal presentations. Researchers on the WIP project even explicitly concluded that various discourse models such as RST relations (see e.g. Mann, Matthiessen & Thompson, 1992), focus space and reference scope are all applicable to the analysis of text-picture documents and take on extended meanings. This conclusion is consistent with our theoretical assumptions outlined in Section 2 above and with work applying systemic-functional accounts developed for language to semiotic systems other than language. Further, we showed in Section 4 how both linguistic resources and graphic resources can be integrated into a single semiotic system.

5.2 Representation of Image Information

Ideally this survey should cover the representation of a wide range of non-linguistic information, eg. image (including pictures, paintings, photos, drawings and maps), music, and video. However for the purpose of this paper, we will concentrate on the representation of image, i.e. on high-level vision representation.

Klinger and Pizano (1989) propose a set of methods for standardising image-indexing. The methods include identifying structures in visual data , mapping out the interface between visual images and languages. The interface between visual information and language facilitates the set-up of principles for storing and retrieving images in image databases. It is easy to see that language plays a key role in their methods.

Knowledge representation is a fundamental aspect of research in computational vision. Havens and Mackworth (1983) attempt to use schemata as a unifying knowledge representation formalism to support high-level vision analysis (ie. scene analysis versus shape recognition). They identify some general properties of the formalism for representing visual knowledge. For example, visual meaning ("knowledge") must be organised to reflect its natural patterning. Ideally this organisation should include a taxonomic hierarchy moving in delicacy from general constraints valid for almost all visual domains to very specialised constraints associated with specific visual objects and their configurations. The processes that operate on knowledge modules must be local to particular visual entities. The representation scheme for visual knowledge they propose bears a clear similarity to the systemic organisation of the meaning potential that we have proposed in this paper.

The strong association between high-level vision and language led Fukuyama and Sugeno (1995) to use linguistic semantics in vision understanding and decision-making that involves perception. In their system, visual data are translated into qualitative and quantitative linguistic specifications using fuzzy sets. These specifications can then serves as a resource for reasoning in combination with information from other domains.

5.3 Integration of multimodal information

One of the most challenging aspect of multimodal systems is to integrate information expressed in various forms, from various semiotic systems, into a common meaning base so that unified reasoning about the information can take place. This problem is actually shared across several engineering fields. In particular, the study on decision making has long investigated the problem under the heading of **data fusion**. For example, Waltz and Buede (1986) applied data fusion techniques to assist combat decision-making; information from a variety of sensors and sources is fused together and interpreted accordingly in the construction of a best possible estimation of a military situation.

A novel approach to integrating diversified information, known as **information fusion with natural language** was proposed by Sugeno (1993) and has been implemented successfully by Kobayashi (Sugeno & Kobayashi 1994; Kobayashi 1995) in a system that predicts the fluctuation of foreign exchange rates. Kobayashi's system accepts input from a wide range of information sources that potentially contributes to the change of the foreign exchange rate; for example: major economic indices, information from a database of current exchange rate from banks, statements by "VIPs". It reconstrues the information from these diverse sources as linguistic meaning (using fuzzy sets), and then applies heuristic-based estimation models on the linguistic meaning to produce plausible estimates of changes. The fundamental assumption behind the notion of "information fusion with natural language" is that language provides the overarching meaning potential which subsumes the information construed by other semiotic systems. This assumption is consistent with the approach we have explored here.

6. Conclusion

The first part of this paper sketched a theoretically unified approach to multimodality, based on systemic-functional theory, as it has been applied to language as well as to other semiotic systems. The second part of this paper presents a case study of multimodal environmental situation assessment (specifically the example of weather forecasting) drawing on the theoretical framework introduced in the first part. We briefly characterised the linguistic and the cartographic meaning systems within the register of standard weather reports, and showed in detail how to develop an integrated meaning potential from this characterisation. We also commented on content planning for a computational model to produce multimodal weather reports. A great obstacle in dealing with multimodal input is to translate image information into symbolic representation such that it is subject to conventional symbolic reasoning. Since image understanding is a challenging issue itself, we stipulate that all visual information to the system also be represented symbolically if it is to be manipulated by the system[5]. Moreover, we are working with the premise that all symbolic representations are to be translated into a natural language based semantic representation. Such a premise is feasible because language provides an overarching semantic space within which various other semiotic systems can find a

---

[5]Some researchers actually attempt to translate visual information into natural language information so that it can be reasoned uniformly with all other forms of information.

corresponding place. In engineering terms, the process based on this premise is necessary because various information resources with diversified forms need to be reasoned with uniformly by the text planner in order to produce a coherent text with complementary multiple modalities. On the basis of the integrated meaning potential, the planning mechanism can thus achieve a high level of uniformity.

We hope to have suggested the practical value of adopting an approach based on a unified and comprehensive theory for all semiotic systems involved in a multimodal system. The alternative would be to adopt a more eclectic approach, selecting different types of account for different domains of the multimodal system. However, we believe that our approach makes it easier to place the different semiotic systems relative to one another in the overall semiotic space they make up and also to reason about similarities and differences, both in the modelling of the multimodal system and in the generation of multimodal presentations (cf. Matthiessen & Nesbitt, in press, on the contrast between theoretical unity and theoretical eclecticism in linguistics).

References

Appelt, D. 1985. *Planning English Sentences.* Cambridge: Cambridge University Press.

Arens, Y. Miller, L., Shapiro, S. and Sondheimer, N. 1988. Automatic Construction of User-interface Displays. *Proceedings of the 7th AAAI Conference*, St. Paul, MN.

Feiner S. K. and McKeown K. R. 1989. Coordinating Text and Graphics in Explanation Generation. *Proceedings Speech and Natural Language Workshop*, Massachusetts. San Mateo: Morgan Kaufmann, 424-433.

Fukuyama, Y. and M. Sugeno. 1995. "A Study on Data Fusion based on Systemic Functional Linguistic Theory." *Proceedings of the International Joint Conference of the 4th IEEE International Conference on Fuzzy Systems.*

Halliday M. A. K. 1966. Lexis as a Linguistic Level. In C. E. Bazell, J. C. Catford, M. A. K. Halliday and R. H. Robins (eds.) *In Memory of J. R. Firth.* London: Longmans, 148-162.

Halliday M. A. K. 1976. Theory. In G. R. Kress (ed.) *Halliday: System and Function in Language.* London: O.U.P., 33-98.

Halliday, M.A.K. 1978. *Language as social semiotic. The social interpretation of language and meaning.* London: Edward Arnold.

Halliday M. A. K. 1985. *An Introduction to Functional Grammar.* London: Edward Arnold.

Halliday, M.A.K. 1995. Fuzzy grammatics: a systemic functional approach to fuzziness in natural language. *Proceedings of FUZZ-IEEE/IFES '95* (International Joint Conference of the Fourth IEEE International Conference on Fuzzy Systems and the Second International Fuzzy Engineering Symposium, Yokohama, 20-24 March 1995).

Halliday M. A. K. and Hasan R. 1985. *Language, Context, and Text: Aspects of Language in a Social-Semiotic Perspective.* Geelong, Victoria: Deakin University Press.

Halliday M. A. K., A. McIntosh and P. Strevens, 1964. Linguistic Sciences and Language Teaching. London: Longmans.

Halliday, M.A.K. & C. Matthiessen. 1994. Construing experience: a language-based approach to cognition. Draft, to appear: Berlin: de Gruyter.

Havens, W. and A. Mackworth. 1983. "Representing and Using Knowledge of the Visual World." in Cercone, N. and McCalla, G. (eds) *The Knowledge Frontier: Essays in the Representation of Knowledge.* Berlin: Springer Verlag.

Hovy E. H. 1989. Unresolved Issues in Paragraph Planning. In C. Mellish and R. Dale (eds.) *Selected Papers from 2nd European Workshop on Language Generation*, Edinburgh, 1989.

Kerpedijev, S. 1992. "Generation of Multimodal Weather Reports." in Dale, R., Hovy, E., Rosner, D. and Stock, O. (eds) *Aspects of Automated Natural Language Generation, Lecture Notes in Artificial Intelligence, N. 587.* Berlin: Springer Verlag.

Klinger, A., and A. Pizano. 1989. "Visual Structure and Data Bases." in Kunii, T. (eds) *Visual Database Systems.* Elsevier Science Publishers.

Kobayashi, I. 1995. *Social System Simulation based on Human Information Processing.* Ph.D dissertation, Department of System Science, Tokyo Institute of Technology.

Mann, W., C. Matthiessen & S. Thompson. 1992. Rhetorical Structure Theory and Text Analysis. In Mann & Thompson (eds), *Diverse descriptions of a text.* Amsterdam: Benjamins.

Matthiessen, C. 1993. Register in the round: diversity in a unified theory of register analysis. In M. Ghadessy (ed.), *Register analysis. Theory and practice.* London: Pinter.

Matthiessen, C. 1995. *Lexicogrammatical cartography: English systems.* Tokyo: International Language Sciences Publishers.

Matthiessen, C. in press. Theme as an enabling resource in ideational knowledge construction. In M. Ghadessy (ed.), *Thematic developments in English texts.* London: Pinter.

Matthiessen, C. & C. Nesbitt. in press. On the idea of theory-neutral description. In D. Butt., C. Cloran & R. Hasan (ed.), *Functional descriptions — linguistic form and linguistic theory.* London: Longman.

McKeown, K. 1982. *Generating Natural Language Text in Response to Questions about Database Queries.* Ph.D. dissertation, University of Pennsylvania.

McKeown K. R. 1985. *Text generation.* Cambridge: Cambridge University Press.

Sacerdoti, E. 1977. *A Structure for Plans and Behaviour.* Amsterdam: Elsevier-North Holland.

Sugeno, M. 1993. "Toward Intelligent Computing." *Proceedings of the 5th IFS World Congress*, Seoul 1993.

Sugeno, M. and I. Kobayashi. 1994. "An Approach to Social System simulation based on Linguistic Information: an Application to the Forecast of Foreign Exchange Rate Changes." *Japanese Journal of Fuzzy Theory and Systems*, Volume 6, Number 4, 1994. (English version has been published by Allerton Press, New York).

Wahlster, W., E. Andre, W. Finkler, W. Graf, H. Profitlich, T. Rist, and A. Schauder. 1992. "WIP: Integrating Text and Graphics Design for Adaptive Information Presentation." In Dale, R., Hovy, E., Rosner, D. and Stock, O. (eds) *Aspects of Automated Natural Language Generation, Lecture Notes in Artificial Intelligence, N. 587*, Berlin: Springer Verlag.

Waltz, E. and Buede. 1986. In IEEE transactions on systems, man and cybernetics, Vol. SMC-16, No. 6.

Towards Detecting Patterns of Human Behaviour from Image Sequences

Robin Hosie
Svetha Venkatesh
Geoff West

Computer Science Department
School of Computing
Curtin University of Technology
PO Box U1987
Perth, WA 6001
Australia
email: robin/svetha/geoff@cs.curtin.edu.au

1   Abstract

This paper details research that will explore the analysis of human behaviour via video surveillance. Digital computer images will be obtained from video footage of a real world scene, and positions of people in the scene will be identified and tracked through each frame in the sequence.

The noted positions will build into a pattern of motion that can be examined and classified. It is proposed that specific events, such as panic or fight situations, will have unique, and therefore identifying, characteristics that will enable automatic detection of such events.

It is envisaged that active cameras will be used when a situation of interest occurs, to enable more information to be extracted from the scene (e.g., panning to follow action, or zooming to enhance detail).

2 Introduction

There is increasing interest in the use of video technology for surveillance in buildings and open spaces, both private and public. Current use can be defined as video recorder based or manual monitoring based.

The aim of this project is to investigate the application of computer vision and machine learning techniques to advancing the performance of surveillance systems. Such systems are mainly used to analyse the actions of human subjects in a scene to determine the legality of actions, detecting dangerous situations, monitoring crowd scenes etc.

A video-based system uses recorders to store low quality images from one or more cameras enabling eight hours or more of surveillance to be recorded on one tape. Some systems decrease the frame sampling rate to enable a longer time period of capture to be condensed onto the tape. These video-based systems are used 'after the event' to determine what happened possibly many hours later. Manual monitoring occurs where one or more people view a number of monitors (up to forty separate cameras on some systems) and respond to some event by zooming in for a better view, recording the relevant action, or even calling the police. Both of these methods of operation have limitations which can be overcome by adding some intelligence to the systems. A simple example of improvement is the addition of some means to detect movement in a scene so video recording only occurs when there is something happening.

Relevant issues are the number of cameras required to cover an area and the resolution required. Given that 100% coverage of an area is required, there are a number of solutions. One is to have special lenses that give a camera a very wide field of view. A low cost, compact, low resolution, integrated lens and CCD array has recently been developed at Edinburgh University. However such a camera has a very poor resolution resulting in poor identification of people in the scene. A second solution is to use a large number of cameras to get high resolution and comprehensive coverage although this is expensive and each camera can only observe a small area. A third solution is the use of actively controlled cameras that can be rotated (pitch and yaw) and zoomed. This allows one or more cameras to *selectively* acquire information about people, crowds, etc. The current state of the art with computer-based techniques means this approach currently applies only to manual monitoring.

We propose to investigate the use of computer vision and artificial intelligence techniques to automate and improve the performance of the surveillance process. The issues that need to be addressed are:

- The development of robust techniques to extract the positions of people in scenes, both static and moving.

- The development of algorithms to track people as they move through scenes.

- The development of methods to control the camera parameters depending on the task and context to maximise the useful information extracted from the images

e.g. to zoom in to record the face of a person.

- The development of techniques to describe the activities of single people and groups in typical situations (e.g. crowd scenes).

- The development of algorithms to determine, given the camera specifications and environment model, the optimum positions and number of cameras needed to monitor an area.

Overall, we shall explore the use of controllable cameras to analyse scenes and interpret the actions of various groups of people. The aim is to be able to determine if people are acting normally or fighting, if there is a robbery in progress, if someone is in need of medical attention (collapsed on a bench, say), etc.

3 Background

Active Vision

The majority of research and development into computer vision, machine vision and image processing has been based on one major assumption, namely processing one or more images from a fixed camera position (Ballard and Brown, 1982; Shirai, 1987). Although this approach has led to a number of successful solutions, these have only been successful in a limited number of domains, for example industrial inspection (West et al., 1991) and remote sensing. The reason for their success is that the problem domain is well defined and constrained, e.g. 2D images of essentially 2D scenes, fixed lighting, etc. When considering the major problem of understanding unstructured environments, there has been little progress in the development of robust solutions.

In a new paradigm (Bajcsy, 1988), active or animate vision systems are proposed that have active control of camera parameters such as focus, zoom, aperture and orientation. In essence, the idea is to move the cameras and camera platform in a similar way to the movement of the eyes and head of a person. This allows the system to selectively sense in space, resolution and time by changing the camera parameters or the data processing technique. Active vision simplifies the processing both in early visual computation (by removing the need to process the whole view at a high resolution) and in higher level visual computation (by movement of the camera around an object to disambiguate solutions).

Although some interest has been shown in the active vision paradigm (Aloimonos et al., 1988; Ballard, 1991; Krotkov, 1989), there are a relatively small number of researchers working in the area mainly because of the need to build specialised hardware, namely a camera platform or vision head. In fact the importance of active vision was revealed by the organisation of a special meeting held in the USA (Swain and Stricker, 1991). The main recommendations of this meeting was that active vision was the most important

direction for computer vision and that one of the major problems is the lack of a standard, low cost, active vision head or camera platform.

There are a number of research issues in active vision, namely attention, foveal sensing, gaze control, gaze stabilisation and gaze change. In essence the research issues are (1) where to look to satisfy particular objectives and (2) how to control the hardware at a high enough speed to obtain real time operation.

Surveillance is an ideal application area for active vision as a single camera cannot give the resolution and field of view to satisfy the requirements.

Surveillance

The analysis of motion characteristics of a single person has been the focus of great attention in recent years. Hogg (1983) was one of the first to detect a walking person by using simple image processing techniques and a powerful geometric reasoning engine, and this work has been extended to adapting the model used for detection to allow for the non-rigidity of human shape in motion (Baumberg and Hogg, 1994). The fundamental advantage of this technique is that the system *learns* the allowable changes in shape for walking people. Images of human behaviour have been studied to detect and identify activities that exhibit regular cyclic characteristics such as skipping or walking (Polana and Nelson, 1993; Polana, 1994). Also, analysing gait to aid recognition of individuals has been explored by Rohr (1993) and Niyogi and Adelson (1994).

All of these approaches are concerned with detecting and examining the motion attributes of an individual, and they use clear images where a single person is relatively easy to isolate. However, in many situations where surveillance is useful, there is often more than one person to consider, creating problems of correspondance and occlusion.

To track multiple objects, Liou and Jain (1991) consider the entire sequence as a 3D volume and group in spatio-temporal space to extract qualitative information about the motion of several objects, while Bouthemy and Lalande (1990) use statistical models to track objects and handle occlusion. In later work, Meyer and Bouthemy (1994) propose a generalised tracking method that is performed in two successive stages: detection and discrimination of objects of interest, and then tracking of those targets. Regions of interest are segmented based on motion with respect to a stationary camera, and tracked through the sequence based on affine models of motion and region geometry. This method handles partial occlusion and large interframe motion in a robust manner, but is restricted to rigid objects.

Surveillance for intruder detection near high security establishments has been explored by Rosin and Ellis (1991). In this work, simple processing of sequences of images was used to extract 'blobs' (a small usually shapeless region of the image) and the movement of these blobs analysed using a frame-based AI system. The system could differentiate between flocks of birds, small animals and other organisms in poor quality images.

Rao *et al.* (1993) outline a surveillance system using several cameras and a decentralised topology to monitor a factory room in real-time. To track people in the scene, the system uses a partitioned Kalman filter and minimal inter-camera communication in order to make each camera an intelligent sensing node. Multiple target tracking is handled using a modified nearest neighbour algorithm, which is computationally cheap but suboptimal in cases where occlusion occurs.

Howarth (1994) looks at analysis and reasoning of automotive traffic. This is analogous to the proposed research in that it is interested in tracking multiple objects of interest through a known scene, and determining trajectories in order to detect events and reason about behaviour. However, extending this idea to the analysis of human traffic adds complexity in two important areas:

1. Humans are non-rigid objects which makes segmentation and tracking more difficult. In general, tracking algorithms rely on rigid objects that do not change shape (just position and possibly pose) from one frame to the next. People in motion exhibit change in shape as well, and research specifically targeted to detecting human motion includes determining and learning flexible models (Baumberg and Hogg, 1994) and time averaging of optic flow (Shio and Sklansky, 1991).

2. In the automotive traffic domain, precise road rules apply that allow labelling abnormal behaviour as any deviation from standard expected patterns. For example, cars should always drive on the left, and turn to the left on entering a round-about. Any object that does not adhere to these rules can be defined as behaving abnormally.

   Some work has been undertaken to analyse behaviour of a human group in domains where rules of behavior do apply: Kawashima *et al.* (1994) uses colour information to discriminate between two soccer teams and describe qualitatively the states of play, while Intille and Bobick (1995) incorporate contextual information to track gridiron players. However, in a general situation of human traffic, such as a hotel foyer, or a shopping mall, less rigid, if any, rules apply.

Most of these systems use fixed cameras for which the parameters remained constant. There has been little work on the active control of cameras for surveillance work. However, there has been work on the area of tracking in which the camera parameters are controlled to keep a moving subject in the field of view (Ballard, 1988; Fermuller and Aloimonos, 1993; Curwen *et al.*, 1992). All these systems use very expensive high performance active vision heads that are uneconomic for surveillance except in very high security areas.

An issue of importance to surveillance is where to position the cameras. Sensor placement has been addressed (Cowan and Kovesi, 1988; Cowan, 1991) mainly in the context of automatic inspection for which the exact geometry of the object is available. However, these techniques have not been applied to surveillance.

4 Proposed Paradigm

The aim of this research is twofold: to investigate techniques to be incorporated into surveillance applications by using active vision methods, and to analyse behaviour patterns in order to be able to automatically detect certain patterns of motion that correspond to certain events.

Current systems used in surveillance typically use one static camera. To cover a wide field of view, these systems have low resolution images, leading to poor performance and ambiguity in image analysis. We propose to use multiple cameras to monitor a scene, where each camera has four degrees of freedom: zoom, pan, tilt and focus. In multiple camera scenarios each camera can survey part of the environment at a higher resolution than provided by a single wide angle camera, and zoom in to investigate moving objects of interest in the image.

Current surveillance systems are also manually monitored, requiring dedicated human attention to view screens or footage, which can be very tiring. Automatic detection of events of interest will serve to highlight situations where closer monitoring is warranted, or action is to be taken.

The numerous issues already stated can be combined into the following three stages:

1. Blob tracking and blob extraction in a single camera viewframe.

2. Tracking blobs from one camera viewframe to the next.

3. Using machine learning to analyse and detect patterns of moving blobs.

Stage 1 involves the extraction and tracking of blobs acquired from a single camera position, that is, from a single viewframe. This involves analysis of the image to determine where to look or investigate next. We propose to investigate both cartesian and log-polar techniques (Rojer and Schwartz, 1990) to guide the camera to foveate on regions of interest. Log-polar analysis gives variable resolution sensing and Lim *et al.* (1995) have developed techniques to detect moving objects in the low resolution periphery and to guide the high resolution fovea to the region of interest.

Stage 2 is concerned with one of the crucial areas in using multiple cameras for surveillance, namely the need to establish correspondence of a blob between two camera viewframes. This will effectively mean that the blob can be tracked from camera to camera. Important issues to be addressed are the need to do this for varying camera orientations and to overcome the problems of tracking multiple blobs. This requires the development of techniques to find correspondences between the various blobs visible in the many camera viewframes, given the known values or ranges of camera parameters. This also requires that the issue of planning is addressed. For example: should the camera parameters be changed (zoom, pan, tilt), should the system concentrate on only one blob, etc.

Stage 3 involves the use of machine learning for analysing the patterns of moving blobs. An important aspect of surveillance is identifying when certain patterns of human behaviour are *interesting* and need further data acquisition, a closer look, or an alarm sounding. Examples of human behaviour that are important in surveying street scenes include identifying fights, robberies, drug peddling, accidents, people being chased etc. An initial examination of the movements of people in crowds reveals the similarity of the movement to the flocking of birds or fish for which graphical models based on simple rules have been produced. We propose to investigate the use of in-house machine learning techniques, such as condition rule generation (Pearce *et al.*, 1994) which uses unary and binary attributes. For example, unary features are the blobs and the binary features are the relationships between pairs of blobs. As such, the learning will act as *inverse flocking* in that given the pattern of moving blobs, the flocking rules will be determined and detected.

It is envisaged that classification of types of behaviour will be possible by examining the interaction between people in the scene. In order to distinguish *abnormal*[1] patterns of behaviour that indicate an unusual event in the scene, it will be first necessary to determine the bounds of *normal* behaviour[2], which will obviously differ from location to location, and even between particular times of the day. For example, human traffic flow in a shopping mall during the day would exhibit different behaviour patterns from those at a hotel lift lobby at night. It may be necessary to calibrate the surveillance system to monitor a particular location, and learn the characteristics of both normal and abnormal events for that specific situation.
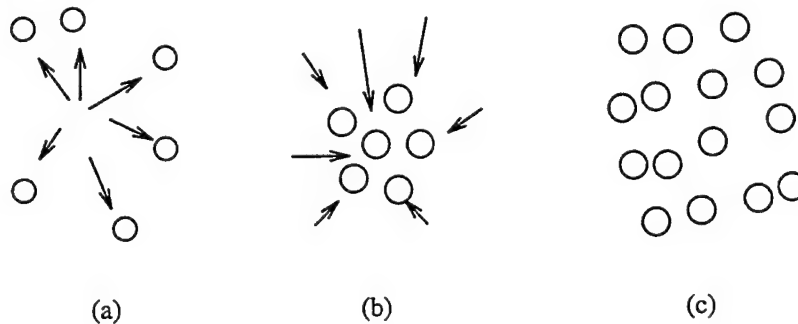

(a) (b) (c)

Figure 1: Patterns of possible abnormal group behaviour.

However, there may be some universal indicators that denote events that are interesting in a wide variety of locations. As an example, people diverging from or converging to a central point at a higher than usual velocity will possibly indicate an event of interest, such as a fight or disturbance, at the point of divergence or convergence (Fig. 1a and b). Similarly, people remaining stationary, or almost stationary, for long periods of time in sensitive areas may give rise to suspicion (Fig. 1c).

The results of this research are expected to be applied to the problem of intelligent surveillance for many applications such as city centres, shopping centres etc. As such,

---

[1] Abnormal in this context is taken to mean unusual occurances.

[2] Ditto usual occurances.

it will be used to extend the performance of current surveillance systems and reduce the human monitoring requirements for such systems.

5 Preliminary Results

Preliminary work has begun in the early stages of the surveillance system described above. Camera and video equipment that are already in place in the foyer areas of the building housing the School of Computing, Curtin University, provide the image sequences to be analysed.

The system, at this stage, is not concerned with real-time processing and the sequences are digitised from previously acquired video footage. The sequence frames are converted into separate images in Silicon Graphics greyscale rgb format, and then smoothed to reduce noise. Currently, segmentation is achieved through simple differencing from a background image, and is constrained to only one person in the scene at any time (at present). This restriction has allowed faster commencement of work in spatial representation and position detection.

The lighting in the foyer scenes is largely uncontrolled with fluorescent lights overhead, and ambient outdoor light from the right of the image (see Fig. 2). There are two cameras situated in the ceiling corners on the west wall, but currently only one view per sequence is considered. The images shown in this paper are from the north-west camera.
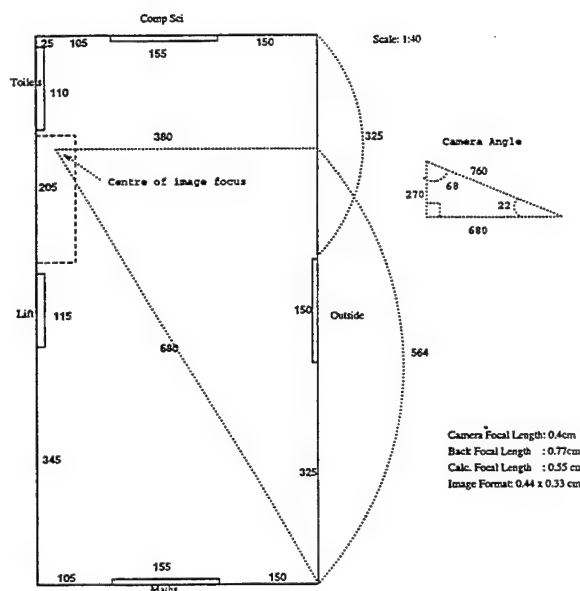

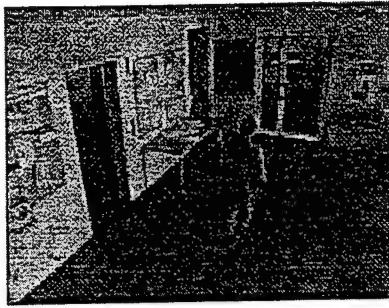Figure 2: Plan of foyer under surveillance.

Figure 3: Unprocessed image from foyer sequence.

Figure 4: Smoothed image.

Foyer sequence

In this sequence a person moves from the North (lower middle of the images), and traverses the foyer to head towards the corridor in the upper left of the image. There were 40 images in this sequence.

Fig. 3 shows a typical image from the foyer sequence, which was smoothed (Fig. 4) using a Gaussian filter.

The difference image (Fig. 5) was generated by differencing (within a threshold) from a background image with pixel intensity equal to the mode intensities over the image sequence. Note that creation of the background image should occur regularly to compensate for the change in outside light over time. As the light in the scene can cause shadows falling on the foyer wall to be included in the segmented, difference image, it is necessary to narrow the search space for locating blobs that may be people by considering only those points that fall in the area of the floor.

The position of the person is noted (Fig. 6) by a simple blob search, which is pruned to consider only points that are contained within the floor polygon. This restriction reduces the chance of mistaking a shadow falling on the white wall as a person. Currently the position of the person is noted as the point in the blob that is 'lowest' in the image, i.e., closest to the bottom edge of the image, as this corresponds to the person's feet in the real world. This location technique is only a rough guide to the position in the real world, and it is intended to be enhanced when more precise position data is required.

As the camera is angled to the foyer floor, it is desirable to rectify each image frame so that they overlay onto a scaled image of the foyer. This enables reasoning about position and trajectory to be simplified to the 2D case, in coordinates that are known and measured. Fig. 7 shows the background (empty) image that has been rectified through perspective transformation, rotation, translation and scaling, and then combined with a scaled foyer image. Fig. 8 shows an image of the rectified marked position
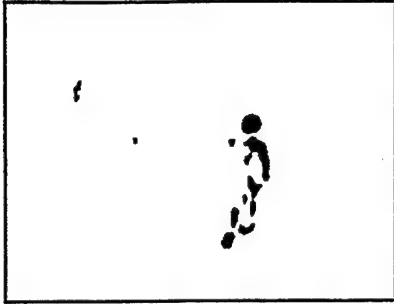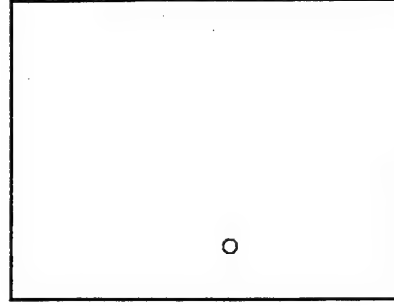
Figure 5: Differenced image.
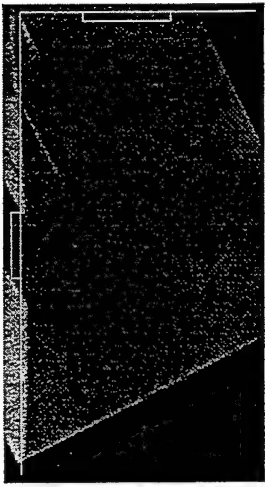

Figure 6: Position marked image.

Figure 7: Rectified background.

Figure 8: Transformed position.

Figure 9: Path on floor plan.

from Fig. 5, and Fig. 9 shows the positions for the entire sequence.

As can be seen in Fig. 9, one person walking through the scene gives rise to a large number of blobs that can give a general indication of the path taken. Examining the inter-blob distances and direction over the image sequence can also give indication of velocity and trajectory respectively.

6 Future Work

The next stage in the research will involve qualitative reasoning about positioning in the scene. It is intended that the floor plan be divided into regions, such as the *lift-area* and *maths-door* so that determination of a person's trajectory will become a matter

of analysing the movement from one region to another. This is intended to lead to classification of those paths most commonly traversed in the scene. For example, the path shown in Fig. 9 may be *maths-door -> central-floor -> table -> compsci-door*.

The appropriate resolution of the floor plan divisions will be determined by analysis, but hierarchical resolution from coarse to fine will enable varying degrees of position accuracy as required.

Currently only one camera view is considered (the north-west camera). It is intended that similar analysis can be applied to the south-west camera sequences, and integrating the two so that a person can be tracked from one camera to another.

7   Conclusion

Video surveillance is gaining increasing interest in the community as a deterrent to crime in public and private areas. Current passive surveillance systems require human monitoring in order to detect suspicious behaviour in the surveyed scene, and this can be tedious and time-consuming. Adding a degree of intelligence to these systems can reduce some of the mundane viewing required by alerting the monitor on the automatic detection of certain events. Similarly, the use of active vision in video surveillance also assists the monitor by automatically directing the camera at items of interest and zooming to capture finer detail.

Intelligent video surveillance has been applied to the automotive traffic domain with effective results, but extending the application to human traffic adds layers of complexity that must be overcome to afford any degree of success.

References

Aloimonos, J. Y., Weiss, I., and Bandyopadhyay, A. (1988). Active vision. *International Journal of Computer Vision*, pages 333–356.

Bajcsy, R. (1988). Active perception. *Proceedings of the IEEE*, 76(8), 996–1005.

Ballard, D. H. (1988). Eye fixation and early vision: Kinetic depth. In *2nd International Conference on Computer Vision, Florida, USA*, pages 524–531.

Ballard, D. H. (1991). Animate vision. *Artificial Intelligence*, 48, 57–86.

Ballard, D. H. and Brown, C. M. (1982). *Computer Vision*. Prentice-Hall.

Baumberg, A. and Hogg, D. (1994). Learning flexible models from image sequences. In *Proceedings of the 3rd European Conference on Computer Vision*, pages 299–308. Springer-Verlag.

Bouthemy, P. and Lalande, P. (1990). Detection and tracking of moving objects based on a statistical regularization method in space and time. In *Proceedings of the 1st European Conference on Computer Vision, Antibes, France*, pages 307–311.

Cowan, C. K. (1991). Automatic camera and light-source placement using CAD models. In *First IEEE Workshop on Directions in CAD-based Vision, Maui, USA*, pages 22–31.

Cowan, C. K. and Kovesi, P. D. (1988). Automatic sensor placement from vision task requirements. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **10**, 407–416.

Curwen, R., Blake, A., and Zisserman, A. (1992). Real-time visual tracking for surveillance and path planning. In *Proceedings of the 1st European Conference on Computer Vision, Santa Margherita Ligure, Italy*, pages 879–883.

Fermuller, C. and Aloimonos, Y. (1993). The role of fixation in visual motion anaylsis. *International Journal of Computer Vision*, **11**, 165–186.

Hogg, D. (1983). Model-based vision: A program to see a walking person. *Image and Vision Computing*, **1**(1), 5–20.

Howarth, R. J. (1994). Spatial representation, reasoning and control for a surveillance system.

Intille, S. S. and Bobick, A. F. (1995). Visual tracking using closed–worlds. Technical report, MIT Media Laboratoy.

Kawashima, T., Yoshino, K., and Aoki, Y. (1994). Qualitative image analysis of group behaviour. In *IEEE Conference on Computer Vision and Pattern Recognition, Seattle, USA*, pages 690–693. IEEE.

Krotkov, E. (1989). *Active computer vision by cooperative focus and stereo*. Springer Verlag.

Lim, F.-L., West, G. A. W., and Venkatesh, S. (1995). Feature detection and recognition in active vision utilizing foveation in log polar space. In *Proceedings of the Asian Conference on Computer Vision, Singapore*.

Liou, S.-P. and Jain, R. C. (1991). Qualitative motion analysis using a spatio-temporal approach. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 726–727. IEEE.

Meyer, F. G. and Bouthemy, P. (1994). Region-based tracking using affine motion models in long image sequences. *CVGIP*, **60**(2), 119–140.

Niyogi, S. A. and Adelson, E. H. (1994). Analyzing and recognizing walking figures in XYT. In *IEEE Conference on Computer Vision and Pattern Recognition, Seattle, USA*, pages 469–474. IEEE.

Pearce, A., Caelli, T., and Bischoff, W. F. (1994). Rulegraphs for graph matching in pattern recognition. *Pattern Recognition*, **27**(9), 1231–1247.

Polana, R. and Nelson, R. (1993). Detecting activities. In *IEEE Conference on Computer Vision and Pattern Recognition, New York, USA*, pages 2–7. IEEE.

Polana, R. B. (1994). *Temporal Texture and Activity Recognition*. Ph.D. thesis, University of Rochester.

Rao, B., Durrant-Whyte, H., and Sheen, J. (1993). A fully decentralized multi-sensor system for tracking and surveillance. *The International Journal of Robotics Research*, **12**(1), 20–44.

Rohr, K. (1993). Incremental recognition of pedestrians from image sequences. In *IEEE Conference on Computer Vision and Pattern Recognition, New York, USA*, pages 8–13. IEEE.

Rojer, A. S. and Schwartz, E. L. (1990). Design considerations for a space-variant sensor with complex-logarithmic geometry. In *10th International Conference on Pattern Recognition, Atlantic City, USA*, volume D, pages 278–285. IEEE.

Rosin, P. L. and Ellis, T. J. (1991). Frame-based system for image interpretation. *Image and Vision Computing*, **9**, 353–361.

Shio, A. and Sklansky, J. (1991). Segmentation of people in motion. In *MOTION91*, pages 325–332.

Shirai, Y. (1987). *Three-dimensional computer vision*. Springer-Verlag.

Swain, M. J. and Stricker, M. (1991). Promising directions in active vision. Technical Report CS 91-27, Department of Computer Science, University of Chicago, 1100 E. 58th Street, Chicago, Illinios 60637, USA.

West, G. A. W., Fernando, T., and Dew, P. M. (1991). CAD based inspection: Using a vision cell demonstrator. In *First IEEE Workshop on Directions in CAD-based Vision, Maui, USA*, pages 155–164. IEEE.

Computer-Generated Forces and Agent-Oriented Technology*

Gil Tidhar
Australian Artificial Intelligence Institute
Level 6, 171 LaTrobe Street
Melbourne 3000, Australia
gil@aaii.oz.au

Graeme Murray
Air Operations Division, DSTO
G.P.O. Box 4331
Melbourne 3001, Australia
Graeme.Murray@dsto.defence.gov.au

Serena Steuart
Air Operations Division, DSTO
G.P.O. Box 4331
Melbourne 3001, Australia
Serena.Steuart@dsto.defence.gov.au

November 1, 1995

"Outside, it's pitch black, no moon and a heavy overcast sky has completely obliterated the meagre night illumination....It is not the sort of night you would like to be out driving your car, but there you are at 60 meters above the ground travelling at close to 1,000 kph. You're thinking to yourself, 'the most intelligent decision I could have made was to stay at home'...."

WGCDR Rick Owen, Royal Australian Air Force

1 Introduction

The modelling and simulation of war like scenarios is one of the prime methods used by analysts for evaluating the effectiveness of different tactics and equipment. Using computer-generated forces is a cheap and safe way of investigating such scenarios.

One of the biggest challenges that face analysts involved in simulating military operations and researchers in artificial intelligence is the modelling and simulation of humans that participate in these scenarios. These include the modelling of combat pilots, ship captains, and generals. In this paper we describe how current agent-oriented technology can be used in generating artificial forces and in particular the modelling and simulation of humans that participate in military operations. We analyze the required behaviour and features of a modelling system and discuss the benefits and limitations of using such agent-oriented technology.

The behaviour exhibited by humans in war-like scenarios is very complex. This behaviour can be described in an abstract way as comprising the following four steps: (1) perceive the world; (2) identify the situation; (3) choose an appropriate response; and (4) act.

Perceiving the world may involve receiving perceptual input from multiple sensors and fusing these inputs into a single model of the world (i.e., *sensor* or *data fusion*). Identifying the situation may involve reasoning about the model of the world provided by the sensors and ascribing additional (subjective) knowledge to parts of the perceived world. Choosing the appropriate response may involve selecting a tactic, deciding to create a plan of attack, etc. Acting involves execution of the chosen response, be it the performance of a tactic, issuing a command, or modifying the sensors.

On the one hand all of these mental and physical activities are performed in real-time and require the ability to react to changing circumstances while responses are being selected and executed. On the other hand humans have particular characteristics that affect their behaviour: they may have varying levels of experience and knowledge, they have emotions such as fear and determination, they have limited capacity for performing complex computations, and they are limited in their physical abilities.

The concept of agent-oriented programming has recently been receiving increasing attention within the artificial intelligence and software engineering communities [9]. Such interest has also contributed to the development of agent-oriented systems [2, 5] and the use of such systems in the area of simulation [7, 11, 14].

Agent-oriented systems seem to be naturally suited to the modelling and simulation of the reasoning processes that are performed by humans. Before we describe how such systems can be used to build simulation systems we would like to describe in some detail the required behaviour from a system that simulates humans in war-like scenarios. In Section 2 we describe these requirements and in Section 3 we provide an analysis of the way agent-oriented systems can best be used. In Section 4 we provide a description of one particular agent-oriented system and in Section 5 we describe how this system has been used for modelling air combat missions. We conclude in Section 6 with a short discussion.

2   Required Behaviour

As mentioned earlier in this work we only focus on modelling and simulation of humans that participate in war-like scenarios; we ignore other aspects such as modelling the dynamics of equipment. Tambe et al. [12] have provided some insight into building believable agents for simulation environments. Here we provide more detailed requirements and identify four types of requirements necessary for a model of human behaviour: (1) ability to interact with the environment; (2) ability to exhibit rational behaviour when reasoning about the world; (3) ability to exhibit irrational behaviour ; and (4) ability to provide a good simulation environment. A model of a human involved in war-like scenarios should thus be able to satisfy all four requirements. A basic aspect of human behaviour is the way humans interact with the environment. These interactions are achieved through a variety of sensors and actuators. We thus require that the simulation system include the following features:

1. **Sensing:** The ability to sense the world through multiple sensors, e.g., eyes, ears, etc., and create a single model of the world from multiple sensory input.

2. **Actions and Physical Capabilities:** The ability to act and affect the world, e.g., push a button, talk, etc. and to conform to physical limitations as determined by the human body.

When reasoning about the world humans use a variety of reasoning techniques and methods. These include situation awareness, planning, pursuing multiple goals simultaneously, and inter-leaving goal-driven and data-driven behaviours. We thus require that the simulation system also include the following features:

3. **Situation Awareness:** The ability to analyze the model of the world and identify particular aspects that require a response.

4. **Decision Making and Reasoning:** The ability to perform complex reasoning, e.g., make decisions, plan, perform spatial and temporal reasoning, etc.

5. **Simultaneous Goals:** The ability to hold multiple goals simultaneously and interleave the achievement of these goals.

6. **Goal-Driven and Data-Driven:** The ability to react to the changing world and to interleave pursuing goals (goal-driven behaviour) and reacting to the world (data-driven behaviour), e.g., react to a missile heading towards the aircraft while deciding which aircraft should be attacked first.

Humans exhibit behaviours which are not always rational and do not always correspond to a prescribed behaviour. Thus a model of human behaviour should be able to simulate emotions, social awareness, and innovation. We thus require that the simulation system also include the following features:

7. **Emotions:** The ability to represent and manipulate emotions and model the way these emotions affect other processes, e.g., the ability to react, the ability to make decisions, etc.

8. **Social Awareness:** The ability to interact with other humans being modelled and to represent and manipulate social structures, e.g., exhibit team behaviour, have commitment towards other team members, etc.

9. **Innovation:** The ability to adopt innovative and novel responses when faced with unfamiliar scenarios, e.g., a pilot may use a combination of tactics in a different way than during training.

The above requirements relate to the fidelity of the simulation. As these models are typically used for the purpose of analysis and evaluation of war-like scenarios there are additional requirements from such models. These requirements refer to the simulation environment itself and include the following features:

10. **Determinism and Repeatability:** Given a particular scenario, the ability to always exhibit a predetermined behaviour and the ability to repeat the exact simulated behaviour under similar conditions.[1]

11. **High Level Specifications:** The ability to specify and modify the behaviour of the agent using a high-level and relatively abstract language.

---

[1] It may be required that the model exhibit a particular probabilistic behaviour, e.g., turn left 80% of the time, but then the determinism and repeatability should be measured using statistical analysis over repeated simulations.

12. **Explanations:** The ability to provide clear and high-level explanations as to the way the reasoning has been performed.

13. **Levels of Knowledge:** The ability to model a variety of types and levels of knowledge - including both knowledge about the world (i.e., descriptive knowledge) and knowledge on how to behave in the world (i.e., procedural knowledge).

14. **Real-Time Performance:** The ability to perform the simulated activities in time frames comparable to human performance, e.g., react within a few tens of milliseconds, evaluate the situation within a few seconds, etc.

The details of the models developed will depend on the required fidelity of the simulation and particular aspects of the scenario that are being investigated. For some investigations it may be sufficient to include a crude model of the behaviour or to ignore some aspects altogether.

3   Using Agent-Oriented Systems

In this section we describe which of the above models can be best implemented using agent-oriented systems. As with any approach or system the agent-oriented approach is not well suited to all types of problems and domains. Nevertheless, it is suitable for the modelling of human behaviour and in particular the modelling of the reasoning performed by humans as described above.

3.1   What Current Agent-Oriented Systems Offer

Current artificial intelligence techniques used for generating models that simulate (or emulate) human expert behaviour can be classified under two approaches. The first is the generation of artificial expert behaviour using training or learning techniques, e.g., Artificial Neural Networks [4]. The second is the generation of artificial expert behaviour using a formal model of this behaviour that is represented either explicitly in declarative plans or implicitly in a reasoning engine, e.g., agent-oriented systems [3, 5].

The major advantage of using the first approach is that there is no need to have a formal understanding of the expert behaviour, and examples combined with expert feedback can be used to train the system. The major disadvantage is that the system can not provide explanation as to the behaviour it exhibits and repeatability is not always guaranteed. The advantages (or disadvantages) of one approach are also the disadvantages (or advantages) of the other approach. The choice of the approach will depend on the particular problem and domain.

It follows from the requirements related to the simulation environment (requirements 11–14) that the model used should include an explicit and well understood formulation of the modelled behaviour. Thus agent-oriented systems seem the most suitable choice for building a simulation system used for investigating military operations. When analyzing agent-oriented systems in the context of the requirements detailed above we identify a major feature of these systems: the specification of an agent's behaviour is primarily based on the concept of a plan.

A plan is an abstract combination of sub-goals to be achieved and actions to be taken. Such plans can either be generated on the fly using a reasoning engine (i.e., a planner) or can be specified in advance in plan libraries. Typically in agent-oriented systems the language used to describe such plans is a high-level language (requirement 11) that allows the analyst to have

a better understanding of the way the agent behaves and the reasons for the choices it makes (requirement 12).

Plans are reasoned about and executed using some form of a reasoning engine that is capable of performing complex reasoning, e.g., spatial and temporal reasoning, and follow some decision making procedure, e.g., means-ends analysis (requirement 4). The nature and complexity of the reasoning is a combination of the engine itself and the plans it manipulates. These could be modified as required to allow the agent to exhibit varying levels of knowledge and capabilities (requirement 13). The speed of the generated response will depend on the complexity of the plan and the reasoning. Requiring that abstract plans be provided in advance and that they be combined during simulation to form the appropriate response allows for the system to perform under real-time constraints (requirement 14).

Since the behaviour of the agent is totally dependent either on the declarative knowledge provided in the plans or on the algorithm of the planner then the behaviour of the system will be totally deterministic. If the simulation of the dynamics of the scenario is also deterministic then the whole simulation will be repeatable (requirement 10).

Current agent-oriented systems are basically centered around a variety of Belief, Desire, Intention (BDI) models [5, 8]. The explicit representation of the desires (or goals) and the intentions of the agent also allows the agent to maintain multiple simultaneous goals (requirement 5). This feature combined with the continuous interleaving of sensing, reasoning, and acting ensures that the agent both reacts to the changing world and interleaves goal-driven and data-driven behaviours (requirement 6). As to the process of situation awareness (requirement 3) it seems that as the level of understanding of this mental process increases so does the ability to provide a formal model of it using agent-oriented systems.

A high-level representation of beliefs and knowledge allows the agent to reason about data as well as abstract concepts (requirement 13). In addition it allows the agent to represent social concepts such as teams, social structures, and roles within a social structure (requirement 8). Other social phenomena such as power structures and informal structures within an organization are still under investigation.

3.2 What Current Agent-Oriented Systems Can't Offer

As mentioned above, although agent-oriented systems can exhibit a very complex behaviour, the current nature of these systems requires that the behaviour be explicitly specified. This implies that they can not actually exhibit behaviour that is not well understood and they can not follow procedures that are not clearly defined. Furthermore, such an approach does not lend itself to performing complex transformations of data (or numbers) or the filtering of such data (or numbers).

Such are the characteristics of some of the required behaviours specified above. In particular it seems that current agent-oriented systems will not be very effective in performing sensing (requirement 1) and incorporating a model of emotions (requirement 7).[2]

Another required behaviour which current agent-oriented systems are unable to provide is innovative behaviour (requirement 9). This limitation goes together with the requirement for real-time performance (requirement 14). The reason is that current technology, and our level of understanding and models of how humans invent novel responses, are limited and they are therefore almost impossible to simulate in real-time.

---

[2]There are two areas of research in artificial intelligence that seem to be particularly well suited towards solving the problem of data and sensor fusion: Fuzzy Logic [15] and Artificial Neural Networks [4].

As mentioned above the characteristics of the specification language and the reasoning engines that execute and manipulate these specifications in agent-oriented systems make them well suited for simulating human reasoning. By the same token, the characteristics of the dynamics of physical systems and the way actions taken affect the world (requirement 2) make agent-oriented systems unsuitable for simulating them.

4 An Agent-Oriented System

In this section we describe one particular agent-oriented system, the dMARS system [2]. It is a distributed multi-agent reasoning system and it provides a representational framework and reasoning mechanisms for implementing agents.

Each agent is composed of a set of beliefs, goals, plans, and intentions. The beliefs of dMARS agents provide information on the state of the environment as perceived by the agent and are represented in a first-order logic. The goals of dMARS agents are descriptions of desired tasks or behaviours.

Plans are declarative procedural specifications that represent knowledge about how to accomplish given goals or react to certain situations [2, 3]. Each plan consists of a body, an invocation condition, and a context condition. The set of plans in a dMARS application system also includes meta-level plans, that is, information about the manipulation of the beliefs, goals, and intentions of the dMARS agent itself.

The body of a plan can be viewed as a procedure or a tactic. It is represented as a graph with one distinguished start node and one or more end nodes. The arcs in the graph are labeled with the sub-goals to be achieved in carrying out the plan. The invocation condition describes the events that must occur for the plan to be executed. Usually, these events consist of the acquisition of some new goals (in which case, the plan is invoked in a goal-directed fashion) or some change in system beliefs (resulting in data-directed invocation) and may involve both. The context condition describes contextual information relevant for the execution of the plan.

The intention list contains all those tasks that the system has chosen for execution, either immediately or at some later time. An intention consists of some initial plan, together with all the sub-plans that are being used in attempting to execute that plan successfully. At any given moment, the intention list of an agent may contain a number of such intentions, some of which may be suspended or deferred, some of which may be waiting for certain conditions to hold prior to activation, and some of which may be meta-level intentions. Only one intention can be executed at any given moment and the choice of that intention depends on the perceived state of the world and the priority of that intention.

In some applications, it is necessary to monitor and process many sources of information at the same time, e.g., simulating a number of pilots. To facilitate this, dMARS was designed to allow several agents to run in parallel. Although the perceptual input received by each agent may come from the same physical world, each agent has its own database, goals, and plans, and reasons asynchronously relative to other agents, communicating with them by sending messages.

5 An Agent-Oriented Air Mission Model

The Smart Whole AiR Mission Model (SWARMM) is an agent-oriented based simulation system developed for the Air Operations Division (AOD) of the Australian Defence Science and Technology Organization (DSTO) and is capable of simulating the dynamics of whole air missions

and the pilot reasoning involved in such missions, and providing a visualization of the simulated mission [1].

The SWARMM system is an integration of three independent models, each implemented using different approaches. Simulating the dynamics of air missions is achieved using a Fortran based system, PACAUS [10], developed by AOD. Simulating the pilot reasoning involved in such missions is achieved using the dMARS system described above. The visualization of the simulated mission is achieved using a 3D graphical system, COMBAT [6], developed by AOD.

The aircraft in air operations are identified by unique names (e.g., SHOGUN-1). Aircraft are teamed together into pairs, groups, and packages and again each team has a unique name (e.g., SHOGUN). As the name *pair* indicates, pairs are teams of two aircraft. Groups are teams made up of pairs and singletons. Packages are teams made up of groups and/or pairs and singletons. For each such team the various teams of which it is made up are referred to as its *sub-teams*.

Each of the sub-teams is assigned at least one role in the team.[3] The role identifies the sub-team's relationships with other sub-teams and the responsibilities it has towards the various functions of the team. We identify two types of structures that are imposed on the team and that correspond to two types of roles. The first is an organizational structure that defines the Command and Control functions in the team, and which is completely hierarchical. We refer to the roles in this structure as *organizational roles*. The second is a functional structure that defines the functional expertise and responsibilities in achieving the task that the team is set to achieve, and does not incorporate any notion of hierarchy. We refer to the roles in this structure as *functional roles*. This model of a team is similar to the model described by Tidhar [13]. Due to the dynamic nature of the domain, teams can be dismantled and re-formed dynamically. As teams change their structure(s) in response to the situation, roles can be dynamically re-assigned.

In dMARS the agent's beliefs are implemented as relations (or predicates) in a relational database. Since each sub-team in a team has at least one role the team itself can be represented as a relation between the team, the sub-teams, and the role that the sub-team has been assigned in the team. We refer to this relation as role-in-team. The only teams that do not have sub-teams are aircraft. Such teams are identified with the predicate singleton (e.g., (singleton SHOGUN-1)).

Tactics are modelled as plans within the dMARS environment. Each agent has plans which define the tactics available to the agent. In order to model coordinated team tactics, plans are written as sets defining the procedures to be used by each member (or agent) within the team. Each agent of the team executes the portion of the tactic which is relevant to itself. The plans or portion of a plan which must be executed by each team member can be differentiated through the context or by branching within the plan. For example, if a team plan has two members (leader and wingman), the part of the plan that is relevant to the leader can be determined by testing if the current agent is the leader in the context condition or branching within the body of the plan.

Each agent is aware of its name and can hence deduce its membership in different teams. Basically, if the agent has been assigned a role in a team then it is a member of that team. If that team has been assigned a role in another team then the agent is also a member of the other team. Not only can the agent deduce its membership in a team, it can also deduce the roles it has been assigned in that team. This knowledge allows the agent to adapt its behaviour as specified in the team tactics.

---

[3] Each sub-team may be assigned more than one role, but it is the responsibility of the designer to ensure that a sub-team is not assigned two conflicting roles.

6 Concluding Remarks

The modelling and simulation of military operations is one of the prime methods used by analysts in evaluating the effectiveness of different tactics and equipment, and in identifying critical problems. Using computer-generated forces is a cheap and safe way of investigating such scenarios.

In this paper we have focused on the modelling and simulation of humans that participate in such scenarios and analyzed the requirements of such a model. We identified four types of requirements: (1) ability to interact with the environment; (2) ability to exhibit rational behaviour when reasoning about the world; (3) ability to exhibit irrational behaviour ; and (4) ability to provide a good simulation environment.

By analyzing the features of current agent-oriented systems in the light of these requirements we have identified that such systems are well structured towards the simulation environment and that they are best suited to modelling the (rational) reasoning processes. Such reasoning processes include making complex decisions, holding and achieving multiple goals simultaneously, reacting to changes in the world while achieving goals, and reasoning about the current state of the world.

With respect to irrational processes that affect human behaviour it seems that given our current level of understanding of such processes current agent-oriented systems can only provide a crude and incomplete model. It may be the case that as the level of understanding of such irrational processes increases so may the ability of agent-oriented systems to model this behaviour. This will depend on the characteristics and details of such processes.

We have provided a short description of the dMARS system which is one particular implementation of the procedural reasoning approach. We have also demonstrated how this system has been used in conjunction with other systems in building SWARMM. SWARMM is a state-of-the-art air combat simulation environment developed for the Air Operations Division of the Australian Defence Science and Technology Organization. The SWARMM system is currently in its final development stages.

As to future work, it seems that there are two aspects of modelling human behaviour using agent-oriented systems that show promise. These are the modelling of situation awareness and the modelling of social awareness and social interactions. It is intended that as part of the development of the SWARMM system and the supporting foundational research, agent-oriented based models will be developed to allow the modelling and simulation of these processes.

Acknowledgments

References

[1] D. Appla, M. Selvestrel, C. Heinze, D. McIlroy, G. Tidhar, and A. Rao. Smart Whole AiR Mission Model (SWARMM): Technical description. Personal Communication, Air Operations Division, Aeronautical Research Laboratory, Melbourne, Australia, 1994.

[2] Australian Artificial Intelligence Institute, Melbourne, Australia. *The dMARS V1.6.7 System Overview*, September 1995.

[3] M. P. Georgeff and A. L. Lansky. Procedural knowledge. In *Proceedings of the IEEE Special Issue on Knowledge Representation*, volume 74, pages 1383–1398, 1986.

[4] S. Grossberg. *Neural Networks and Natural Intelligence*. MIT Press, Cambridge, MA, 1988.

[5] J. E. Laird, A. Newell, and P. S. Rosenbloom. SOAR: An architecture for general intelligence. *Artificial Intelligence*, 33(1):1–64, 1987.

[6] R. Muscat. Combat display program: Specification and user guide. Personal Communication, Air Operations Division, Aeronautical Research Laboratory, Melbourne, Australia, 1993.

[7] A. Rao, A. Lucas, D. Morley, M. Selvestrel, and G. Murray. Agent-oriented architecture for air combat simulation. Technical Note 42, Australian Artificial Intelligence Institute, Melbourne, Australia, April 1993.

[8] A. S. Rao and M. P. Georgeff. Modelling rational agents within a BDI-architecture. In J. Allen, R. Fikes, and E. Sandewall, editors, *Proceedings of the Second International Conference on Principles of Knowledge Representation and Reasoning*, pages 473–484. Morgan Kaufmann Publishers, San Mateo, CA, 1991.

[9] Y. Shoham. Agent-oriented programming. *Artificial Intelligence*, 60(1):51–92, 1993.

[10] S. Steuart. PACAUS: overview and source code description. Personal Communication, Air Operations Division, Aeronautical Research Laboratory, Melbourne, Australia, 1993.

[11] M. Tambe, W. L. Johnson, E. M. Jones, F. Koss, J. E. Laird, P. S. Rosenblum, and K. Schwamb. Intelligent agents for interactive simulation environments. *AI Magazine*, 16(1):15–39, 1995.

[12] M. Tambe, R. M. Jones, J. E. Laird, P. S. Rosenbloom, and K. Schwamb. Building believable agents for simulation environments: Extended abstract. In *Collected Papers of the SOAR/IFOR Project*, pages 78–81. Information Sciences Institute, University of Southern California, Marina del Rey, CA, 1994. Also published as technical report ISI/SR-94-379.

[13] G. Tidhar. Team oriented programming: Preliminary report. Technical Report 41, Australian Artificial Intelligence Institute, Melbourne, Australia, 1993.

[14] G. Tidhar, M. Selvestrel, and C. Heinze. Modelling teams and team tactics in whole air mission modelling. In G. F. Forsyth and M. Ali, editors, *Proceedings of the Eighth International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems*, pages 373–381, Melbourne, Australia, June 1995. Gordon and Breach Publishers.

[15] R. R. Yager and L. A. Zadeh. *An Introduction to Fuzzy Logic Applications in Intelligent Systems*. Kluwer, Boston, MA, 1992.

Applications of Abduction #3: "Black-Box" to "Gray-Box" Models

Tim Menzies [*]   Simon Goss [†]

November 6, 1995

Abstract

Military Operations Research makes extensive use of large complex simulation models. These simulation models are often *black-boxes*; i.e. they are opaque and incomprehensible. A considerable effort is involved in commissioning a new model into service. We characterise this activity as the construction of *gray-box* approximate causal models of the functional behaviour of *black-box* simulation software.

While the imported black-box models are typically numeric, determinate and precise, their associated gray-box models are under-specified, indeterminant and vague. Here we explore the use of the HT4 *abductive inference* engine to support the process of using ad-hoc experience with black-box models to construct and maintain partially-specified gray-box models.

1   Introduction

Military Operations Research (OR) makes extensive use of large complex simulation models. These can be the result of many person-years of development and incorporate modules obtained from external sources. It is common that such a system may be obtained from a third party. These simulation models are often *black-boxes*; i.e. they are opaque and incomprehensible. A considerable effort is involved in commissioning a new model into service. In the process, familiarity and expertise is gained in the model. The black-box nature of these simulation systems complicates their verification and validation for local conditions. Customisation

is also difficult for the same reason.

We characterise this activity as the construction of *gray-box* approximate causal models of the functional behaviour of the black-box simulation software. Such gray-box models have several advantages. They can serve to *document* and *preserve* the expertise gained in the pain of commissioning the system. Further, such gray-box models can *inspected, verified* and *validated*, thus increasing our confidence in the results obtained using the software. Finally, such gray-box models can simplify *customisation* of legacy systems.

While the imported black-box models are typically numeric, determinate and precise, their associated gray-box models are under-specified, indeterminant and vague. Yet these gray-box models represent our best understanding of the innerworkings of complex black-box models. Here we explore the use of the HT4 [22, 24] *abductive inference* engine to support the process of using ad-hoc experience with black-box models to construct and maintain partially-specified gray-box models.

This paper is structured as follows. Section 2 of this document reviews the problem of commissioning a model obtained from a remote site to produce a list of requirements on a methodology for validating black-box models. Section 3 describes our preferred abductive framework. Section 4 argues that the requirements in section 2 can be meet by the HT4 abductive inference engine. Section 5 discusses related work in the qualitative reasoning, abductive, and truth maintenance literature. The conclusion discusses further work.

Note that portions of this work have appeared previously (see [24]).

2   Using Remote Models

The remote model commissioning problem is summarised in Figure 1. Some group called TEAM-1 derives some model $\mathcal{M}_1$ representing their *initial understanding* of a problem (e.g. modeling the

---

[*]Dept. of Software Development, Monash University, Caulfield East, Melbourne, VIC, Australia, 3145; +61-3-903-1033; +61-3-903-1077(fax);
Email: timm@insect.sd.monash.edu.au;
URL: http://www.sd.monash.edu.au/~timm

[†]Defence Science and Technology Organisation- Air Operations Division, PO Box 4331 Melbourne, VIC, Australia, 3001. *Email:* simon.goss@dsto.defence.gov.au

performance of a fighter aircraft). This model is *operationalised* in some third generation language to become M_2. Perhaps an attempt is made to document M_1 in a *manual* M_3. M_1 is commonly a research prototype comprising thousands or hundreds of thousands of lines of code. Hence M_3 is typically incomplete. M_2 and M_3 are then shipped to another site where a second team (TEAM-2) tries to understand them. Conceptually, TEAM-2 builds M_4, a model representing the *local understanding* of M_2 and the incomplete M_3. Current practice is for M_4 to be documented in an incomplete manner (e.g. some procedural manual advising parametric sensitivity and constants relating to the local physical and operational environment).
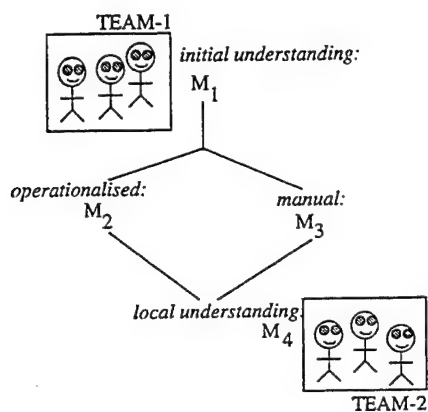

Figure 1: Commissioning Remote Models

The effort required to use M_2 with confidence can be non-trivial. In our experience in the Air Operations Division, the effective operational use of large OR simulation codes can be time consuming. This can be even more time consuming when object code is supplied without source code or access to the author.

In essence M_2 is a *black-box model* that Team 2 must convert (with some support from M_3) into a *gray-box model* M_4. Once validated, M_4 would be used for planning, prediction, and optimisation studies. Note the emphasis on *validation*. Local conditions may invalidate M_2. The Australian Defence Forces (ADF) use aircraft in configurations that are different to how they are used overseas. Certain decision parameters for scenario outcomes are stored in compiled numerical matrices and are inaccessible to TEAM-2 . For example, these parameters may (i) be based on experimental data from tests in other climates or (ii) contain certain tacit assumptions about aircraft operation. Prior

to relying on M_2, TEAM-2 would like to validate this model under local conditions.

Therefore, our desired solution supports:

Requirement 1 *(i) Validation of models; (ii) planning and prediction using the validated model; (iii) generating multiple options from the validated models, from which we can chose the optimum approach.*

The validation module would be particularly important. Each translation from M_i to M_j can introduce errors. Also, even though we imply that the members of TEAM-1 and TEAM-2 have the same model, this may not be the case. Individuals within a team may incorrectly believe they share the same view of a problem. Such a validation engine would allow individuals to check their own model as well as settling disputes between competing models; e.g. the best models have fewer problems.

While we refer to the construction of M_1 and M_4, these models may never be formally recorded. For example, M_4 may only ever be tacit since it is built during the second team's informal conversations about M_2 and M_3. This is a major problem since if staff are transferred, they take their hard-won understanding of M_2 with them. We need to somehow structure the development of M_4 such that the experience gained in this process is not lost. Therefore:

Requirement 2 *An ideal model comprehension tool would be a workbench within which M_4 can be documented.*

Note that TEAM-2 may not be able to communicate openly with TEAM-1 . The company that employs TEAM-1 may have only sold M_2 and M_3 as stand-alone products without any consultancy support. Nor may TEAM-2 have full access to the source code of M_2. For example, legal or contractual obligations of TEAM-1 may prevent disclosure of portions of M_2 to (say) non-US citizens. Such portions may only be available in binary format. Hence, M_4 will be an under-specified "back of the envelope" sort of model containing guesses about the internal structure of M_2. Therefore:

Requirement 3 *The representation system of M_4 must be able to handle under-specified models.*

Such under-specified models are indeterminant at runtime. When competing influences act on the same entity, but the magnitude of these influences is under-specified, then the modeling system

must be able to create one world for each possible outcome. Note that the ability to create multiple worlds also supports the processing of "what-if" scenarios. This is a useful function for models built for exploratory purposes such as $\mathcal{M}_4$.

Assumption management will also be useful when we try to execute the guess that is $\mathcal{M}_4$. Inference over an uncertain model will generate assumptions whenever we traverse some unmeasured portion of the model. Mutually exclusive assumptions must be managed in separate worlds. Therefore:

Requirement 4 *A model comprehension tool should include assumption management and multiple world reasoning.*

3 Abduction

In this section, we discuss an inference procedure called abduction. In the next section we will argue that this procedure can satisfy the requirements of commissioning remote models.

3.1 An Introduction to Abduction

Informally, abduction is inference to the best explanation [30]. Given α , β, and the rule $R_1 : \alpha \vdash \beta$, then *deduction* is using the rule and its preconditions to make a conclusion $(\alpha \wedge R_1 \Rightarrow \beta)$; *induction* is learning R_1 after seeing numerous examples of β and α; and *abduction* is using the postcondition and the rule to assume that the precondition could be true $(\beta \wedge R_1 \Rightarrow \alpha)$ [19].

More formally, abduction is the search for assumptions $\mathcal{A}$ which, when combined with some theory $\mathcal{T}$ achieves some goal $\mathcal{G}$ without causing some contradiction [4]. That is:

EQ_1: $\quad \mathcal{T} \cup \mathcal{A} \vdash \mathcal{G}$

EQ_2: $\quad \mathcal{T} \cup \mathcal{A} \not\vdash \perp$

While abduction can be used to generate explanation engines, we believe that EQ_1 and EQ_2 are more than just a description of "inference to the best explanation". EQ_1 and EQ_2 can be summarised as follows: make what inferences you can that are relevant to some goal, without causing any contradictions. Our basic argument is that that the proof trees used to solve EQ_1 and EQ_2 contain many of the inferences we want to make.

3.2 The HT4 Abductive Inference Engine

In order to understand abduction in more detail, we describe our HT4 abductive inference engine [22, 24]. To execute HT4, the user must supply a theory $\mathcal{T}$ comprising a set of uniquely labeled statements S_x. For example, from Figure 2, we could say that:

```
s[1] = plus_plus(a,b).
s[2] = minus_minus(b,c).
etc.
```

Figure 2 is an under-specified qualitative model [14]. In that figure:

- $X \xrightarrow{++} Y$ denotes that Y being *UP* or *DOWN* could be explained by X being *UP* or *DOWN* respectively;

- $X \xrightarrow{--} Y$ denotes that Y being *UP* or *DOWN* could be explained by X being *DOWN* or *UP* respectively.

Note that the results of this model may be uncertain; i.e. it is indeterminate. In the case of both A and B going *UP*, then we have two competing influences of C and it is indeterminate whether C goes *UP*, *DOWN*, or remains *STEADY*.
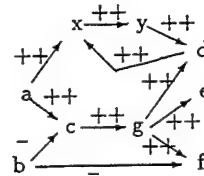

Figure 2: $\mathcal{T}_1$: An indeterminate qualitative model.

The dependency graph $\mathcal{D}$ connecting literals in $\mathcal{T}$ is an and-or graph comprising $<< \mathcal{V}^{and}, \mathcal{V}^{or} >, \mathcal{E}, \mathcal{I} >$; i.e. a set of directed edges $\mathcal{E}$ connecting vertices $\mathcal{V}$ containing invariants $\mathcal{I}$. $\mathcal{I}$ is defined in the negative; i.e. $\neg \mathcal{I}$ means that no invariant violation has occurred (e.g. if $\mathcal{I}(p, \neg p)$, then we block the simultaneous belief in a proposition *and* its negation). Each edge $\mathcal{E}_x$ and vertex $\mathcal{V}_y$ is labeled with the S_x that generated it.

For example, returning to the theory $\mathcal{T}$ of Figure 2, let us assume that (i) each node of that figure can take the value *UP*, *DOWN*, or *STEADY*; (ii) the conjunction of an *UP* and a *DOWN* can explain a *STEADY*; and (iii) no change can be explained in terms of a *STEADY* (i.e. a *STEADY* vertex has no children). With these assumptions,

we can expand Figure 2 into Figure 3. In that figure, $\mathcal{V}^{and}$ vertices are denoted (e.g.) *&002* while all other vertices are $\mathcal{V}^{or}$ vertices. Note that in practice, the assumptions used to convert $\mathcal{T}$ into $\mathcal{D}$ are contained in a domain-specific *model-compiler*.
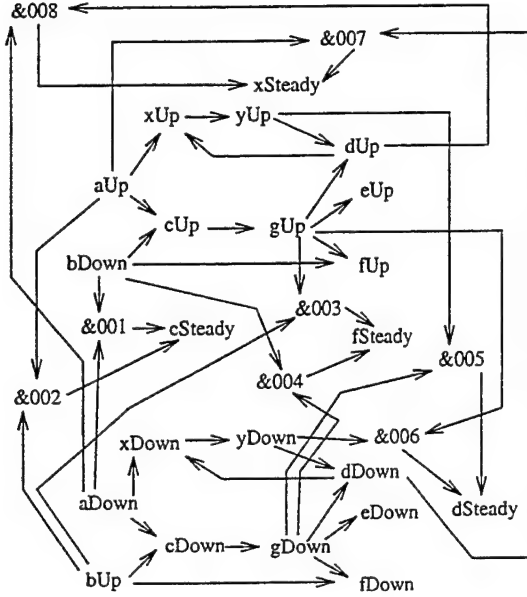

Figure 3: $\mathcal{D}$ calculated from the $\mathcal{T}$ of Figure 2

Not shown in Figure 3 are the invariants. For a qualitative domain, where nodes can have one of a finite number of mutually exclusive values, the invariants are merely all pairs of mutually exclusive assignments; e.g.:

```
i(aUp, aSteady). i(aSteady, aUp).
i(aUp, aDown).   i(aDown,   aUp).
i(bUp, bSteady). i(bSteady, bUp).
i(bUp, bDown).   i(bDown,   bUp).
etc.
```

3.2.1 Using HT4

HT4 extracts subsets of $\mathcal{E}$ which are relevant to some user-supplied $\mathcal{TASK}$. Each $\mathcal{TASK}_x$ is a triple $< \mathcal{IN}, \mathcal{OUT}, \mathcal{BEST} >$. Each task comprises some $\mathcal{OUT}$puts to be reached, given some $\mathcal{IN}$put ($\mathcal{OUT} \subseteq \mathcal{V}$ and $\mathcal{IN} \subseteq \mathcal{V}$). For the rest of this paper we will explore the example where:

```
IN  = {aUp, bUp}
OUT = {dUp, eUp, fDown}
```

$\mathcal{IN}$ can be either be a member of the known $\mathcal{FACTS}$ or a $\mathcal{DEFAULT}$ belief which we can assume if it proves convenient to do so. Typically,

$\mathcal{FACTS} = \mathcal{IN} \cup \mathcal{OUT}$. If there is more than one way to achieve the $\mathcal{TASK}$, then the $\mathcal{BEST}$ operator selects the preferred way(s).

To reach a particular output $\mathcal{OUT}_z \in \mathcal{OUT}$, we must find a proof tree $\mathcal{P}_x$ using vertices $\mathcal{P}_x^{used}$ whose single leaf is $\mathcal{OUT}_z$ and whose roots are from $\mathcal{IN}$ (denoted $\mathcal{P}_x^{roots} \subseteq \mathcal{IN}$). All immediate parent vertices of all $\mathcal{V}_y^{and} \in \mathcal{P}_x^{used}$ must also appear in $\mathcal{P}_x^{used}$. One parent of all $\mathcal{V}_y^{or} \in \mathcal{P}_x^{used}$ must also appear in $\mathcal{P}_x^{used}$ *unless* $\mathcal{V}_y^{or} \in \mathcal{IN}$ (i.e. is an acceptable root of a proof). No subset of $\mathcal{P}_x^{used}$ may contradict the $\mathcal{FACTS}$; e.g. for invariants of arity 2:

$$\neg(\mathcal{V}_y \in \mathcal{P}_x^{used} \wedge \mathcal{V}_z \in \mathcal{FACTS} \wedge \mathcal{I}(\mathcal{V}_y, \mathcal{V}_z))$$

For our example, the proofs are:

```
p(1) = {aUp, xUp,   yUp,   dUp}
p(2) = {aUp, cUp,   gUp,   dUp}
p(3) = {aUp, cUp,   gUp,   eUp}
p(4) = {bUp, cDown, gDown, fDown}
p(5) = {bUp, fDown}
```

3.2.2 Assumptions

The union of the vertices used in all proofs that are not from the $\mathcal{FACTS}$ is the HT4 assumption set $\mathcal{A}_{all}$; i.e.

$$\mathcal{A}_{all} = \left(\bigcup_{\mathcal{V}_y} \left\{ \mathcal{V}_y \in \mathcal{P}_x^{used} \right\} \right) - \mathcal{FACTS}$$

The proofs in our example makes the assumptions:

```
a = {xUp, yUp, cUp, gUp, cDown, gDown}
```

The union of the subsets of $\mathcal{A}_{all}$ which violate $\mathcal{I}$ are the *controversial assumptions* $\mathcal{A}_C$:

$$\mathcal{A}_C = \bigcup_{\mathcal{V}_x} \{ \mathcal{V}_x \in \mathcal{A}_{all} \wedge \mathcal{V}_y \in \mathcal{A}_{all} \wedge \mathcal{I}(\mathcal{V}_x, \mathcal{V}_y) \}$$

The controversial assumptions of our example are:

```
ac = {cUp, gUp, cDown, gDown}
```

Within a proof $\mathcal{P}_y$ the *preconditions* for $\mathcal{V}_y \in \mathcal{P}_x^{used}$ are the transitive closure of all the parents of $\mathcal{V}_y$ in that proof. The *base controversial assumptions* ($\mathcal{A}_B$) are the controversial assumptions which have no controversial assumptions in their preconditions (i.e. are not downstream of any other controversial assumptions). The base controversial assumptions of our example are:

```
ab = {cUp, cDown}
```

3.2.3 Worlds

Maximal consistent subsets of $\mathcal{P}$ (i.e. maximal with respect to size, consistent with respect to $\mathcal{I}$) are grouped together into worlds $\mathcal{W}$ ($\mathcal{W}_i \subseteq \mathcal{E}$). Each world $\mathcal{W}_i$ contains a consistent set of beliefs that are relevant to the $\mathcal{TASK}$. The union of the vertices used in the proofs of $\mathcal{W}_i$ is denoted $\mathcal{W}_i^{used}$. In terms of separating the proofs into worlds, $\mathcal{A}_B$ are the crucial assumptions. We call the maximal consistent subsets of $\mathcal{A}_B$ the *environments* $\mathcal{ENV}$ ($\mathcal{ENV}_i \subset \mathcal{A}_B \subseteq \mathcal{A}_C \subseteq \mathcal{A}_{all} \subseteq \mathcal{V}$). The environments of our example are:

```
env(1) = {cUp}
env(2) = {cDown}
```

The union of the proofs that do not contradict $\mathcal{ENV}_i$ is the world $\mathcal{W}_i$. In order to check for non-contradiction, we compute the exclusions set $\mathcal{X}$. $\mathcal{X}_i$ are the base controversial assumptions that are inconsistent with $\mathcal{ENV}_i$. The exclusions of our example are:

```
x(1) = {cDown}
x(2) = {cUp}
```

A proof $\mathcal{P}_j$ belongs in world $\mathcal{W}_i$ if it does not use any member of $\mathcal{X}_i$ (the excluded assumptions of that world); i.e.

$$\mathcal{W}_i = \bigcup_{\mathcal{P}_j} \left\{ \mathcal{P}_j^{used} \cap \mathcal{X}_i = \emptyset \right\}$$

Note that each proof can exist in multiple worlds. The worlds of our example are:

```
w(1) = {p(1), p(2), p(3), p(5)}
w(2) = {p(1), p(4), p(5)}
```

$\mathcal{W}_1$ is shown in Figure 4 and $\mathcal{W}_2$ is shown in Figure 5.
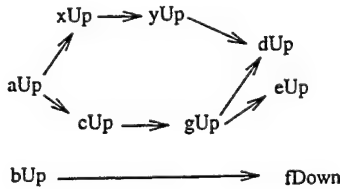

Figure 4: $\mathcal{W}_1$

For any world $\mathcal{W}_i$, $\mathcal{W}_i^{causes}$ are the members of $\mathcal{IN}$ found in $\mathcal{W}_i$ ($\mathcal{W}_i^{causes} = \mathcal{W}_i^{used} \cap \mathcal{IN}$). The achievable or *covered* goals $\mathcal{G}$ in $\mathcal{W}_i$ are
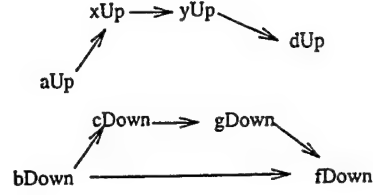

Figure 5: $\mathcal{W}_2$

the members of $\mathcal{OUT}$ found in that world ($\mathcal{W}_i^{covered} = \mathcal{W}_i^{used} \cap \mathcal{OUT}$). Continuing our example:

```
causes(w(1)) = {aUp, bUp}
causes(w(2)) = {aUp, bUp}

cover(w(1))  = {dUp, eUp, fDown}
cover(w(2))  = {dUp, fDown}
```

3.2.4 The $\mathcal{BEST}$ of all Possible Worlds

Note that, in our example, we have generated more than one world and we must now decide which world(s) we prefer. This is done using the $\mathcal{BEST}$ criteria. Numerous $\mathcal{BEST}$s can be found in the literature; e.g. the $\mathcal{BEST}$ worlds are the one which contain:

1. the most specific proofs (i.e. largest size) [8];

2. the fewest *causes* [35];

3. the greatest *cover* [22];

4. the most number of specific concepts [32];

5. the largest subset of $\mathcal{E}$ [29];

6. the largest number of *covered* outputs [28];

7. the most number of edges that model processes which are familiar to the user [31];

8. the most number of edges that have been used in prior acceptable solutions [18];

Our view is that $\mathcal{BEST}$ is domain specific; i.e. we believe that there is no universally best $\mathcal{BEST}$.

4 Abduction and Remote Model Commissioning

In this section we argue that our abductive model can be used to satisfy the requirements of remove

model comprehension; i.e. it can support validation, planning, prediction, optimisation, inference over under-specified models using assumption management and multiple-worlds reasoning.

4.1 Inference Over Under-Specified Models

HT4 can execute over indeterminate/ under-specified models. Further, if this execution generates assumptions, then these assumptions are managed in mutually exclusive worlds $(\mathcal{W})$.

4.2 Validation

Validation tests a model's validity against external semantic criteria. Given a library of known behaviours (i.e. a set of pairs $< \mathcal{IN}, \mathcal{OUT} >$), abductive validation uses a $\mathcal{BEST}$ that favours the worlds with largest number of covered outputs (i.e. maximise $\mathcal{IN} \cap \mathcal{W}_x$) [28].

Note that this definition of *validation* corresponds to answering the following question: "can a model of X explain known behaviour of X?". We have argued elsewhere that this is the definitive test for a model [22]. Note that this is a non-naive implementation of validation since it handles certain interesting cases. In the situation where no current model explains all known behaviour, competing theories can be assessed by the extent to which they cover known behaviour. $\mathcal{M}_X$ is definitely better than $\mathcal{M}_Y$ if $\mathcal{M}_X$ explains far more behaviour than theory $\mathcal{M}_Y$.

As an example of validation-as-abduction, recall that $\mathcal{W}_1$ (see Figure 4) was generated from $\mathcal{T}_1$ when $\mathcal{IN} = \{\text{aUp, bUp}\}$ and $\mathcal{IN} = \{\text{dUp, eUp, fDown}\}$. Note that $\mathcal{W}_1^{covered}$ is all of $\mathcal{OUT}$. $\mathcal{T}_1$ is hence not invalidated since there exists a set of assumptions under which the known behaviour can be explained.

See the related work section for a discussion of other validation approaches.

4.3 Planning

Planning is the search for a set of operators that convert some current state into a goal state. We can represent planning in our abductive approach as follows:

- Represent operators as rules that convert some state to some other state;

- Augment each operator rule with:

  - a unique label $\mathcal{S}_1$, $\mathcal{S}_2$, etc. When $\mathcal{D}$ is generated, each edge will now include the name(s) of the operator(s) that generated it.

  - A cost figure representing the effort required to apply this operator rule.

- Set $\mathcal{IN}$ to the current state, $\mathcal{OUT}$ to the goal state, and $\mathcal{FACTS} = \mathcal{IN} \cup \mathcal{OUT}$.

- Set $\mathcal{BEST}_{PLANNING}$ to favour the world(s) with the least cost. The cost of a world is the maximum of the "proof cost" of each member of $\mathcal{OUT}$. The "proof cost" of $\mathcal{OUT}_i$ is the minimum cost of the proofs that cover $\mathcal{OUT}_i$.

- Run HT4. Collect and cache the generated worlds.

- For each $\mathcal{BEST}$ world, collect all the names of the operators used in the edges of that world. These operators will be in a tree structure that reflects the structure of the $\mathcal{BEST}$ worlds. Report these trees as the output plans.

A related task to planning is *monitoring*; i.e. the process of checking that the current plan(s) are still possible. The worlds generated by the above planner will contain some assumptions. As new information comes to light, some of these assumptions will prove to be invalid. Delete those worlds from the set of possible plans. The remaining plans represent the space of possible ways to achieve the desired goals in the current situation. If all plans are rejected, then run HT4 again with all the available data.

4.4 Optimisation

We view optimisation as planning with a $\mathcal{BEST}$ operator that favours the lower cost world(s).

4.5 Prediction

Prediction is the process of seeing what will follow from some events $\mathcal{IN}$. This can be implemented in HT4 by making $\mathcal{OUT} \subseteq \mathcal{V} - \mathcal{IN}$; i.e. find all the non-input vertices we can reach from the inputs. For prediction, $\mathcal{FACTS}$ should not be $\mathcal{IN} \cup \mathcal{OUT}$ since this will be the entire dependency graph. If the $\mathcal{IN}$ is certain, then $\mathcal{FACTS} = \mathcal{IN}$ (i.e. only the inputs cannot be contradicted). This is a non-naive implementation of prediction since mutually exclusive predictions (the *covered* elements of $\mathcal{OUT}$) will be found in different worlds.

Note that in the special case where:

- $\mathcal{IN}$ are all root vertices in $\mathcal{D}$.
- $\mathcal{FACTS} = \emptyset$
- $\mathcal{OUT} = \mathcal{V} - \mathcal{IN}$

then our abductive system will compute ATMS-style [2] *total envisionments*; i.e. all possible consistent worlds that are extractable from the theory. A more efficient case is that $\mathcal{IN}$ is smaller than all the roots of the graph and some *interesting subset* of the vertices have been identified as possible reportable outputs (i.e. $\mathcal{OUT} \subset \mathcal{V} - \mathcal{IN}$).

5 Related Work

5.1 General Abductive Reasoning

Note that this work is part of Menzies' *abductive reasoning project*. Menzies argues that abduction provides a comprehensive picture of declarative knowledge-based systems (KBS) inference such as prediction, classification, explanation, quantitative reasoning, planning, monitoring, set-covering diagnosis, consistency-based diagnosis, validation, and verification [24]. Menzies also believes that abduction is a useful framework for intelligent decision support systems [23], diagrammatic reasoning [27], single-user knowledge acquisition, and multiple-expert knowledge acquisition [25]. Further, abduction could model certain interesting features of human cognition including the situated nature of cognition [26]. Others argue elsewhere that abduction is also a framework for natural-language processing [29], design [33], visual pattern recognition [34], analogical reasoning [5], financial reasoning [11], machine learning [12] and case-based reasoning [18].

5.2 Qualitative Reasoning

We are not the first researchers to argue that intuitions about models can be represented in an indeterminant, under-specified modeling framework. The qualitative reasoning (QR) community focuses on the processing of systems called qualitative differential equations (QDE) which are:

- Piece-wise well-approximated by low-order linear equations or by first-order non-linear differential equations;
- Whose numeric values are replaced by one of three qualitative states: up, down, or steady [14].

Since QDEs are under-specified, they can be written faster than their fully-specified quantitative counterparts. Hence, they have been proposed as a tool for recording intuitions. However, we do not suggest using QR for building $\mathcal{M}_4$. A QDE is still a mathematical equation and mathematics is a poor model for causality. Ohms's Law ($R = \frac{V}{I}$) relates resistance R to current I and voltage V. Note that changes in voltage and current do not cause changes in resistance, even though the mathematical formulae suggests this is possible. Resistors cannot be manufactured to a certain specification merely by attaching wire to some rig and altering the voltage and current over the rig. Ignoring the effects of temperature and high-voltage breakdown, resistance is an invariant built into the physics of a wire. Hidden within Ohm's Law are rules regarding the direction of causality between voltage, current, and resistance. Such rules are invisible to a mathematical formulation.

Causality was a central concern in QR till the mid-1980s [1] and it is a construct we wish to support in $\mathcal{M}_4$.

> ... It is clear that causality plays an essential role in our understanding of the world ... to understand a situation means to have a causal explanation of the situation [13].

Initially two qualitative ontologies were proposed: DeKleer & Brown's 1984 CONFLUENCES system [3] and Forbus's 1984 qualitative process theory (QPT) [6]. Later work in 1986 recognised that both these systems processed QDEs and a special theorem prover, QSIM, was written by Kuipers especially for QDEs [16]. Compilers were written to covert QPT models into QSIM. Note that the evolution of QR worked down from complex representations (QPT to QSIM to simpler graph-theoretic approach). Kuipers himself now believes that underlying QSIM was a more basic inference process: Mackworth's arc consistency algorithm [17, 21] which is based around a simple graph-theoretic framework (though Mackworth's work can be expressed in a logic framework [20]). Note the evolution of the QR work from complex representations (e.g. QPT) to simpler graph-theoretic approaches.

After an inclusive public debate between public debate in 1986 between the CONFLUENCES approach and a rival theory [15], the term "causality" was avoided by many QR researchers. Forbus's 1992 retrospective on causality and the 1980s QR research is primarily negative:

... In terms of violating human intuitions, each system of qualitative physics fails in some way to handle causality properly. Like (QPT) theory, deKleer and Brown's CONFLUENCES theory... fails to distinguish between equations representing causal versus non-causal laws. Kuipers QSIM contains no account of causality at all [7].

In summary, the 1980s experiment with using QDEs to model causal intuitions failed. We prefer our directed-graph approach since this at least gives us a strong sense of inference direction.

5.3 Truth-Maintenance

Here we have explored a graph-theoretic framework for mon-monotonic logic. An alternative approach is the logic-based approach pioneered by DeKleer's assumption-based truth maintenance system [2]). In his ATMS framework, an inference engine passes justifications to a database which, as a side-effect, would incrementally modify sets of consistent literals storing the root assumptions of different worlds. Forbus & DeKleer proposed this as a general inference procedure for knowledge-based *problem solvers* [7]. We have a similar intuition. However, unlike the ATMS, Menzies does not divide the inference process between an inference engine and an ATMS database. Rather, Menzies argues that a thorough declarative reading of common KBS can be mapped into the world-generation process described in section 3.

In later work, DeKleer linked his approach with Reiter's default logic [36]. An *extension E* of a default theory is a set of literals from the theory which do not violate a set of invariants (called the *justifications*). All formulae whose preconditions (called *prerequisites*) are satisfied by E and whose invariants are consistent with E are also in E. An HT4 world differs from a default logic extension in that the latter is closed under deduction and contains all literals that are consistent with E. HT4's worlds only contain *relevant literals*; i.e. only the literals that are on proofs leading to known *outputs*. HT4 regards full extension generation as wasted computation.

At its core, the ATMS builds the dependency network between literals in a knowledge base and explores this network. Invariant knowledge is maintained such that mutually incompatible subsets of this dependency network are avoided. Such a representation can be used for validation. Thus dependency network can be used to determine inputs that will exercise all branches of the knowledge base. This is the core of the validation systems by Ginsberg [9] and Zlatereva [37]. However, note that once an input suite is inferred, an expert still has to decide what are the appropriate outputs for those inputs. In the case of vague models (where there is no definitive oracle), the correct outputs are unknown. The remove model comprehension problem is a model construction activity and the constructed model is less a picture of a domain than a device for exploring that domain. Asking a member of TEAM-2 for the correct output across an uncertain knowledge base that is being built to explore an area of uncertainty seems, in our view, inappropriate.

We note that HT4 has much in common with the Ginsberg/Zlatereva approaches. All these systems are based on a TMS variant. More precisely, all these systems use some style of non-monotonic logic. We prefer our approach since we believe that our graph-theoretic approach is a more minimal framework than the logic-based style of Ginsberg and Zlatereva. Initially, we found that logic-based approaches to TMS were very complicated. After mapping the TMS process down to a graph-theoretic process, we found the TMS process more approachable and simpler to understand. HT4 could be used in a Ginsberg/Zlatereva style. If we use HT4 to generate all possible worlds, then the roots of those worlds will be test suite inputs that will exercise all branches of the KB. We hesitate to suggest this as standard practice, however, since the generation of all worlds is even slower than HT4 usual practice of generating worlds for the relevant literals (see the discussion of complexity in [22]).

6 Conclusion

There is a pressing need for some methodology to structure the creation and recording of the understanding of remote models; i.e. the generation of $\mathcal{M}_4$. In terms of the computational requirements of $\mathcal{M}_4$, an appropriate modeling language must support:

- Validation;
- Planning;
- Prediction;
- Optimisation;
- Inference over under-specified models;
- Assumption management and multiple-worlds reasoning.

In this paper, we have argued that abduction is a promising approach since it satisfies these criteria.

We have also noted similarities with of the remote model commissioning problem to the QR and TMS literature. While both the QR and TMS literature supply us with insights into our problem, we find the TMS literature more relevant than QR. Potentially fruitful avenues to explore include:

- A *proof-of-concept study* in which a gray-box model is built using our abductive framework from a readily-available dynamic simulation black-box computer game. The advantage of using such a game is that, unlike OR models, it is small enough to explain in a paper. Furthermore, the game would be available to other researchers.

- *Situation awareness*: When faced with a novel domain, people learn models. There are many styles of learning. We conjecture that people learn models to the depth required for some particular purpose. The resulting models are hence approximate. One way to characterise our current proposal is the construction of approximate models gained through incomplete experience of the entity being modeled. We speculate that this represents a form of situation awareness [10].

References

[1] E. Coiera. The Qualitative Representation of Physical Systems. *The Knowledge Engineering Review*, 7:1–23, 1 1992.

[2] J. DeKleer. An Assumption-Based TMS. *Artificial Intelligence*, 28:163–196, 1986.

[3] J. DeKleer and J.S. Brown. A Qualitative Physics Based on Confluences. *Artificial Intelligence*, 25:7–83, 1984.

[4] K. Eshghi. A Tractable Class of Abductive Problems. In *IJCAI '93*, volume 1, pages 3–8, 1993.

[5] B Falkenhainer. Abduction as Similarity-Driven Explanation. In P. O'Rourke, editor, *Working Notes of the 1990 Spring Symposium on Automated Abduction*, pages 135–139, 1990.

[6] K. Forbus. Qualitative Process Theory. *Artificial Intelligence*, 24:85–168, 1984.

[7] K. Forbus. Pushing the Edge of the (QP) Envelope. In B. Faltings and P. Struss, editors, *Recent Advances in Qualitative Physics*, pages 245–261. The MIT Press, 1992.

[8] C.L. Forgy. RETE: A Fast Algorithm for the Many Pattern/Many Object Pattern Match Problem. *Artificial Intelligence*, pages 17–37, 19 1982.

[9] A. Ginsberg. A new Approach to Checking Knowledge Bases for Inconsistentcy and Redundancy. In *Proc. 3rd Annual Expert Systems in Government Conference*, pages 102–111, 1987.

[10] J. Gledhill and S. Goss. Modeling Situation Awareness. In preparation.

[11] W. Hamscher. Explaining Unexpected Financial Results. In P. O'Rourke, editor, *AAAI Spring Symposium on Automated Abduction*, pages 96–100, 1990.

[12] K. Hirata. A Classification of Abduction: Abduction for Logic Programming. In *Proceedings of the Fourteenth International Machine Learning Workshop, ML-14*, page 16, 1994. Also in *Machine Intelligence* 14 (to appear).

[13] Y. Iwasaki. Causal Ordering in a Mixed Structure. In *Proceedings of AAAI '88*, pages 313–318, 1988.

[14] Y. Iwasaki. Qualitative Physics. In P.R. Cohen A. Barr and E.A. Feigenbaum, editors, *The Handbook of Artificial Intelligence*, volume 4, pages 323–413. Addison Wesley, 1989.

[15] Y. Iwasaki and H.A. Simon. Causality in Device Behaviour. *Artificial Intelligence*, 29:3–31, 1986.

[16] B. Kuipers. Qualitative Simulation. *Artificial Intelligence*, 29:229–338, 1986.

[17] B.J. Kuipers. Reasoning with Qualitative Models. *Artificial Intelligence*, 59:125–132, 1993.

[18] D.B. Leake. Focusing Construction and Selection of Abductive Hypotheses. In *IJCAI '93*, pages 24–29, 1993.

[19] H. Levesque. A Knowledge-Level Account of Abduction (Preliminary Version). In *IJCAI '89*, volume 2, pages 1061–1067, 1989.

[20] A. Mackworth. The Logic of Constraint Satisfaction. *Artificial Intelligence*, 58:3–20, 1992.

[21] A.K. Mackworth. Consistency in Networks of Relations. *Artificial Intelligence*, 8:99–118, 1977.

[22] T. J. Menzies and P. Compton. The (Extensive) Implications of Evaluation on the Development of Knowledge-Based Systems. In *Proceedings of the 9th AAAI-Sponsored Banff Knowledge Acquisition for Knowledge Based Systems*, 1995.

[23] T.J. Menzies. Applications of Abduction #1: Intelligent Decision Support Systems. Technical Report TR95-16, Department of Software Development, Monash University, 1995.

[24] T.J. Menzies. Applications of Abduction #2: Knowledge Level Modeling. Technical Report TR95-23, Department of Software Development, Monash University, 1995.

[25] T.J. Menzies. *Principles for Generalised Testing of Knowledge Bases*. PhD thesis, University of New South Wales, 1995.

[26] T.J. Menzies. Situated Semantics is a Side-Effect of the Computational Complexity of Abduction. In *Australian Cognitive Science Society, 3rd Conference*, 1995.

[27] T.J. Menzies and P. Compton. *A Precise Semantics for Vague Diagrams*, pages 149–156. World Scientific, 1994.

[28] T.J. Menzies and W. Gambetta. Exhaustive Abduction: A Practical Model Validation Tool. In *ECAI '94 Workshop on Validation of Knowledge-Based Systems*, 1994.

[29] H.T. Ng and R.J. Mooney. The Role of Coherence in Constructing and Evaluating Abductive Explanations. In *Working Notes of the 1990 Spring Symposium on Automated Abduction*, volume TR 90-32, pages 13–17, 1990.

[30] P. O'Rourke. Working Notes of the 1990 Spring Symposium on Automated Abduction. Technical Report 90-32, University of California, Irvine, CA., 1990. September 27, 1990.

[31] C.L. Paris. The Use of Explicit User Models in a Generation System for Tailoring Answers to the User's Level of Expertise. In A. Kobsa and W. Wahlster, editors, *User Models in Dialog Systems*, pages 200–232. Springer-Verlag, 1989.

[32] D. Poole. On the Comparison of Theories: Preferring the Most Specific Explanation. In *IJCAI '85*, pages 144–147, 1985.

[33] D. Poole. Hypo-Deductive Reasoning for Abduction, Default Reasoning, and Design. In P. O'Rourke, editor, *Working Notes of the 1990 Spring Symposium on Automated Abduction.*, volume TR 90-32, pages 106–110, 1990.

[34] D. Poole. A Methodology for Using a Default and Abductive Reasoning System. *International Journal of Intelligent Systems*, 5:521–548, 1990.

[35] J. Reggia, D.S. Nau, and P.Y Wang. Diagnostic Expert Systems Based on a Set Covering Model. *Int. J. of Man-Machine Studies*, 19(5):437–460, 1983.

[36] R. Reiter. A Logic for Default Reasoning. *Artificial Intelligence*, 13:81–132, 1980.

[37] N. Zlatereva. Truth Mainteance Systems and Their Application for Verifying Expert System Knowledge Bases. *Artificial Intelligence Review*, 6, 1992.

The Challenge of Whole Air Mission Modelling[1]

Graeme Murray, Serena Steuart, Dino Appla, David McIlroy, Clinton Heinze,
Martin Cross, Arvind Chandran and Richard Raszka

Defence Science and Technology Organisation
Air Operations Division
PO Box 4331, Melbourne, Australia 3001
Graeme.Murray@dsto.defence.gov.au

Gil Tidhar, Anand Rao, Andrew Pegler, David Morley and Paolo Busetta

Australian Artificial Intelligence Institute
Level 6, 171 La Trobe Street, Melbourne, Australia 3000
gil@aaii.oz.au

Abstract

Air mission modelling using graphical simulation provides a
powerful means for development and evaluation of tactics. However,
large models are particularly expensive and time-consuming to
maintain and modify. Multi-aircraft full mission man-in-the-loop
simulators will provide an even more complex programming
environment. The dMARS[2] software provides a suitable environment
for the development of an air mission simulation model based on the
concept of rational agents. This approach allows the analyst to work
at a high level, formulating concepts and aims, while keeping the
detailed computer programming hidden.

This research and development project aims to provide the basis for
an advanced multi-aircraft military simulation called the Smart Whole
AiR Mission Model (SWARMM). It is a simulation system that will
be capable of simulating the physics of whole air missions and the
pilot reasoning involved in such missions. AAII and DSTO-AMRL are
the primary participants in this project in the CRC for Intelligent
Decision Systems.

This system will provide DSTO with the ability to rapidly evaluate
and test counter-air tactics for the RAAF. It will provide high-fidelity
simulation of combat aircraft, ground controlled intercept (GCI)
controllers, and surveillance aircraft, advanced reasoning capabilities

[2] dMARS is proprietary software belonging to the Australian Artificial Intelligence
Institute (AAII).

for modelling the pilot's reasoning process, and sophisticated visualisation tools to enable a better understanding of whole air missions. It will enable DSTO to rapidly create and modify tactics used as part of the pilot's reasoning process as well as rapidly setting up a simulation scenario for operational studies.

1. Introduction

A project has been undertaken to extend the multiple-aircraft-engagement capabilities of Australia's PACAUS air combat model to full air mission simulation [1]. This is being achieved by replacing the FORTRAN tactical reasoning code with an artificial intelligence representation of pilot reasoning whilst retaining PACAUS's present modelling of physical systems.

The new model, known as SWARMM (Smart Whole AiR Mission Model) incorporates a tactical reasoning system based upon the dMARS distributed real-time reasoning system developed by the Australian Artificial Intelligence Institute. The tactical reasoning system resolves functionality-related shortcomings of conventional air combat models. It will:

- be much more accessible to pilots;

- better represent tactics knowledge and human decision making;

- allow rapid display, evaluation, and modification of tactics - including complex (team) tactics;

- provide increased flexibility and ability to handle complex situations such as team decision making and variable commitment;

- enable different plan libraries to be used for evaluating potential threats, tactical environments, new equipment, etc.;

- reduce software support costs; and

- allow the system to be used as a basis for tactical expertise training.

2. The Design of SWARMM

The approach which has been taken involves building a powerful pilot reasoning model capable of interfacing with existing systems simulation software. The key is to separate the pilot reasoning model from the physical model and visualisation software (as illustrated in Figure 1). This can be achieved by embedding the new pilot model representation (in dMARS system) into the current simulation environment (the PACAUS system).

Smart Whole AiR Mission Model (SWARMM)

Graphics/
Visualisation

Aircraft physics,
weapons, sensors,
countermeasures,
environment
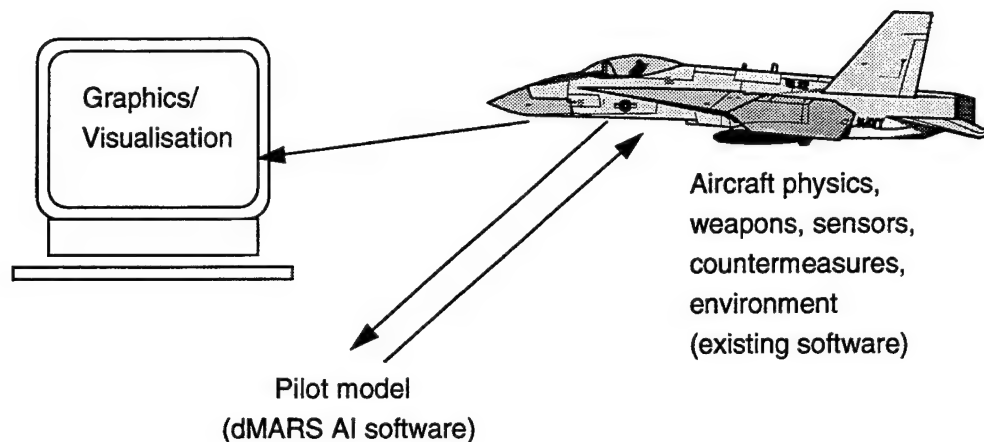(existing software)

Pilot model
(dMARS AI software)

Figure 1

In growing the existing PACAUS model into the SWARMM system we are replacing in PACAUS the existing situation priority reasoning, team cooperation logic and beyond visual range (BVR) tactics suite with new code in the dMARS system; SWARMM will combine time-stepped modelling of physical systems with event-stepped (but time-dependent) pilot reasoning processes.

The structure of PACAUS enables a simple interface between the systems and tactical reasoning models, in effect requiring only the replacement of a subroutine call with a call to dMARS. The interfacing software has to pass messages about the perceived physical world to the reasoning software, and to transmit back instructions for continuing or changing the present action.

The world as perceived by the pilot model is based on the control of information coming from multiple sources. Output from the sensor models can vary from actual world parameters, eg., errors in range or angle measurements. This data is referred to as *sensed world data*. Information provided by different sensors will not necessarily be congruent (particularly in the presence of countermeasures) and the pilot model needs to fuse and utilise all such data to gain an awareness of the world situation (construct the *air picture*). That information (at whatever level of confidence) is used to decide "what will happen if I keep doing what I am doing now", followed by a reaction to improve what is expected to happen.

PACAUS contains routines for aircraft and systems control, such as fighter intercept and combat manoeuvres and the logic of highly dynamic one-on-one close combat counter-manoeuvring. Not transferring this to the reasoning system reduces the amount of information which must be passed through the interface and reduces the dependency of SWARMM on the speed of the decision software. The reasoning system performs the role of tactician (the guy in the back seat) rather than that of the pilot. Tactical instructions are coded as manoeuvres to be flown and, where they are relative to an opponent, indicate which opponent.

This allows the analyst to modify the knowledge base, representing the tactics and decision making processes of pilots, without being concerned about the remaining systems modelling code in PACAUS. Hence, the analyst need only work at the high level, formulating concepts, determining mission goals, and developing pilot tactics. This approach is eminently suited to dealing with extensive repertoires of procedural team tactics.

Simulating the physics of whole air missions involves dealing with multiple aircraft types, multiple roles (strike, escort, sweep, counter-air-defence), multiple weapons systems, multiple sensors and communication systems. The accuracy of the models directly affects the fidelity of the simulation and the effectiveness of using it as a tool for understanding and analysing air missions.

Entering data and mission briefing information will be through graphical user interfaces (GUIs). The majority of this has been completed in a form enabling keyboard input in response to pop-up menus, but this is being progressed into a full GUI for mouse selection of system configurations (aircraft, stores, etc). The mission briefing for each "pilot" (or other operator) is specified by sets of geographic coordinates, goals, teams, packets and a specified plan library (or libraries, appropriately assembled from the general library of plans).

Graphic output will be displayed using the Combat Graphics Software which was written for PACAUS, running as a separate process. This provides great flexibility of presentation and viewpoint, including stereoscopic viewing. Data files can be generated to enable runs to be reviewed in detail.

3. Current Status of SWARMM

Operation has been established between PACAUS and the reasoning software (running on an ONYX and a SUN, respectively) and between PACAUS and the combat graphics package running as separate processes on the same machine; the design allows the graphics to run separately on a networked SGI INDIGO machine.

The testing of a reduced scope scenario has just been completed. The reduced scope scenario involved an analysis of six aircraft, with one team consisting of two aircraft in a Defensive Counter Air (DCA) role, and the other team consisting of two Sweep and two Strike aircraft. The aim of the reduced scope scenario was to enable development, testing and verification of the various aspects of team tactics in teams of up to four aircraft. The number of aircraft in the scenarios is currently being expanded to enable the analysis of mission level operations (up to forty aircraft). The pilot model is being developed to improve the complexity and realism of the simulated pilot responses.

A Scenario Development Language (SDL) has been developed to enable rapid generation of scenarios. The language is used to define parameters such as the number, type and configuration of the aircraft in the scenario. The team structures and hierarchy, and the tactics that will be employed by the teams, subteams and individual aircraft in the scenario are also defined. A Scenario Development Tool (SDT) is used to extract the required information for the scenario, as defined by the SDL, from the appropriate databases and libraries. The extracted information is used to generate the required input data for the pilot model and the physical model.

The Combat Graphics software is being developed to provide options to display mission level operations, and its execution in synchronisation with the model provides capability for man-in-the-loop (MIL) as well as providing essential facilities for observing and verifying the simulation of tactical processes.

4. Future Developments for SWARMM

The reasoning system is being developed also with "hooks" to enable MIL operation. To gain the full benefits of MIL it will be necessary to restructure the present PACAUS software, although it will be possible to patch the existing code for operation with one or two human operators. The benefits of MIL operation have been well established through systems such as the UK Joust facility and the US MIL-AASPEM facilities. One drawback of fully manned simulation facilities is that operational personnel are not freely available to researchers. Incorporating computer generated "players" enables studies of larger scenarios to be undertaken with reduced numbers of pilots. However it has been said of presently available tactical environments that the behaviour of computed opponents and computed associates are identifiable by their "behaviour" and by their predictability, which human participants use to their advantage (resulting in "contamination of study results"). SWARMM is expected to be able to provide computed opponents who display sufficient skill and flexibility to overcome this problem.

5. Conclusion

The approach outlined here makes it easier (and thus faster) to develop and modify tactics (including team tactics) in air mission simulation. It allows tactics to be constructed and displayed graphically (the analyst does not need to program the tactics in source code), separates the tactics from the major body of the simulation code, and makes simulated situations more easily understood by display of the underlying tactics involved.

References

[1] G. Tidhar, M. Selvestrel, and C. Heinze, Modelling Teams and Team Tactics in Whole Air Mission Modelling. In G. F. Forsyth and M. Ali, editors, *Proceedings of the Eighth International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems*, pages 373-381, Melbourne, Australia, June 1995. Gordon and Breach Publishers.

DISTRIBUTION LIST

Proceedings of the Artificial Intelligence in Defence Workshop AI'95

Edited by Simon Goss

AUSTRALIA

1. DEFENCE ORGANISATION

b. S&T Program
Chief Defence Scientist
FAS Science Policy } shared copy
AS Science Corporate Management
Director General Science Policy Development
Counsellor Defence Science, London (Doc Data Sheet)
Counsellor Defence Science, Washington (Doc Data Sheet)
Scientific Adviser to MRDC Thailand (Doc Data Sheet)
Director General Scientific Advisers and Trials/Scientific Adviser Policy and
    Command (shared copy)
Navy Scientific Adviser (Doc Data Sheet and distribution list only)
Scientific Adviser - Army (Doc Data Sheet and distribution list only)
Air Force Scientific Adviser
Director Trials

Aeronautical and Maritime Research Laboratory
Director

Electronics and Surveillance Research Laboratory
Director

CAOD
CMOD
CLOD
CITD
Research Leader Aircraft Performance
Head Theatre Operations Group (K. Anderson)
Simon Goss

DSTO Library
Library Fishermens Bend
Library Maribyrnong
Library Salisbury (2 copies)
Australian Archives
Library, MOD

c. Forces Executive
Director General Force Development (Sea)
Director General Force Development (Land)

d. Navy
SO (Science), Director of Naval Warfare, Maritime Headquarters Annex,
    Garden Island, NSW 2000. (Doc Data Sheet only)

e. **Army**
ABCA Office, G-1-34, Russell Offices, Canberra   (4 copies)
SO (Science), DJFHQ(L), MILPO Enoggera, Queensland 4051 (Doc Data Sheet
    only)
NAPOC QWG Engineer NBCD c/- DENGRS-A, HQ Engineer Centre Liverpool
    Military Area, NSW 2174 (Doc Data Sheet only)

f. **Air Force**
DDSYS (WGCDR P. Viggers) B-1-15, Russell Offices
DDSC1 (WGCDR J. Matthews) A-7-16, Russell Offices
APSC3 (WGCDR B. Sutherland) APSC, RAAF Base Fairbain

g. **S&I Program**
Defence Intelligence Organisation
Library, Defence Signals Directorate (Doc Data Sheet only)

i. **B&M Program (libraries)**
OIC TRS, Defence Regional Library, Canberra
Officer in Charge, Document Exchange Centre (DEC), 1 copy
*US Defence Technical Information Center, 2 copies
*UK Defence Research Information Centre, 2 copies
*Canada Defence Scientific Information Service, 1 copy
*NZ Defence Information Centre, 1 copy
National Library of Australia, 1 copy

2. **UNIVERSITIES AND COLLEGES**

Australian Defence Force Academy
    Library
    Head of Aerospace and Mechanical Engineering
Senior Librarian, Hargrave Library, Monash University
    Librarian, Flinders University

3. **OTHER ORGANISATIONS**

NASA (Canberra)
AGPS
State Library of South Australia
Parliamentary Library, South Australia

OUTSIDE AUSTRALIA

4. **ABSTRACTING AND INFORMATION ORGANISATIONS**
INSPEC: Acquisitions Section Institution of Electrical Engineers
Library, Chemical Abstracts Reference Service
Engineering Societies Library, US
Materials Information, Cambridge Scientific Abstracts, US
Documents Librarian, The Center for Research Libraries, US

5. **INFORMATION EXCHANGE AGREEMENT PARTNERS**
Acquisitions Unit, Science Reference and Information Service, UK
Library - Exchange Desk, National Institute of Standards and Technology, US

DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION DOCUMENT CONTROL DATA	1. PRIVACY MARKING/CAVEAT (OF DOCUMENT)	

2. TITLE Proceedings of the Artificial Intelligence in Defence Workshop AI'95	3. SECURITY CLASSIFICATION (FOR UNCLASSIFIED REPORTS THAT ARE LIMITED RELEASE USE (L) NEXT TO DOCUMENT CLASSIFICATION) Document (U) Title (U) Abstract (U)

4. AUTHOR(S) Edited by Simon Goss	5. CORPORATE AUTHOR Aeronautical and Maritime Research Laboratory PO Box 4331 Melbourne Vic 3001

6a. DSTO NUMBER DSTO-GD-0077	6b. AR NUMBER AR-009-481	6c. TYPE OF REPORT General Document	7. DOCUMENT DATE June 1997

8. FILE NUMBER M1/8/789	9. TASK NUMBER DST 92/091	10. TASK SPONSOR DSTO	11. NO. OF PAGES 147	12. NO. OF REFERENCES 231

13. DOWNGRADING/DELIMITING INSTRUCTIONS None	14. RELEASE AUTHORITY Chief, Air Operations Division

15. SECONDARY RELEASE STATEMENT OF THIS DOCUMENT

Approved for public release

OVERSEAS ENQUIRIES OUTSIDE STATED LIMITATIONS SHOULD BE REFERRED THROUGH DOCUMENT EXCHANGE CENTRE, DIS NETWORK OFFICE, DEPT OF DEFENCE, CAMPBELL PARK OFFICES, CANBERRA ACT 2600

16. DELIBERATE ANNOUNCEMENT

No Limitations

17. CASUAL ANNOUNCEMENT	Yes

18. DEFTEST DESCRIPTORS

artificial intelligence, simulation, operations research, surveillance, modelling, decision support sytems, defence science

19. ABSTRACT

A workshop with the theme "Artificial Intelligence in Defence" was held at the Australian Joint Conference on Artificial Intelligence at the Australian Defence Force Academy in November 1995. There were 52 attendees from defence, defence science, industry and academia. Twelve papers were presented in four thematic areas: Decision Support, Surveillance and Information Fusion, Modelling and Operations Research, and Simulation and Training. This proceedings documents the final versions of those papers.

SPARES (10 copies) 63
ATTENDEES (52 copies)
PAPER AUTHORS (16 copies)

Total number of copies: 131